Model Risk for Exotic Equity Options

MSc Thesis Applied Mathematics

Frank Bervoets
frank.bervoets@rabobank.com
Delft University of Technology, Delft, Netherlands
Rabobank International, Utrecht, Netherlands
September 2006

Model Risk for Exotic Equity Options

Preface

This thesis was submitted in the partial fulfilment for the requirements for the Master's degree in Applied Mathematics at Delft University of Technology. The research for this thesis was conducted during a nine months internship at Rabobank International in Utrecht at the Modelling & Research department.

For about two and a half years I have been part of the Derivatives Research & Validation team of Freddy van Dijk, which is part of the Modelling & Research department. Here I got the opportunity to interchange periods of being a part-time model validator and periods of being a full-time intern. For this, I would like to express my sincere gratitude to Sacha van Weeren and Freddy van Dijk for giving me this great opportunity. I would also like to thank them for the job in their London team. I believe this is a huge career opportunity. Furthermore, I would like to thank my supervisor Roger Lord for proofreading my drafts and for his intensive support. And, of course, my gratitude goes to all other colleagues at Rabobank for giving me this good working environment.

On the University, I want to thank my thesis supervisor Dr. Ir. C.W. Oosterlee for numerous things. First of all, I thank him for introducing me to the world of financial mathematics; next, for the guidance he gave during the last year of my Bachelor and the complete Master; and finally, for proofreading all my drafts, the discussions and the hints he gave. My gratitude also goes to Fang Fang en Coen Leentvaar for all useful discussions and comments, and to Prof. Dr. Ir. P. Wesseling and Dr. C. Kraaikamp for being part of the graduation committee.

Finally, I would like to thank Dr. A. Almendral and Dr. J. Toivanen for allowing me to use their code.

Utrecht, September 2006
Frank Bervoets

Contents

Introduction

In April 1973 derivatives started to be officially traded on the Chicago Board Options Exchange. In that same year Fischer Black and Myron Scholes published the first model based on no arbitrage arguments for pricing options (Black and Scholes, 1973). This model still forms the foundation for modern finance. Certain underlying assumptions are questionable however, e.g. assets are assumed to follow a continuous process and no transaction costs are considered. But most importantly, a drawback is that in the Black-Scholes world the volatility is assumed to be constant. The invalidity of this assumption becomes clear when considering so-called implied volatilities. The implied volatility is that volatility for which the Black-Scholes price equals the market price. Nowadays, the market shows that the implied volatility is dependent on the strike and the maturity of the option.

Since 1973 many other models have been introduced, including stochastic volatility models, jump models and local volatility models. All these models try to capture the flaws of the Black-Scholes model. Some of them have become quite popular and will be discussed in this thesis. These different models also give rise to new problems, e.g. which model is the best or what is the risk of the choice of model? The first problem is difficult, because it is dependent on the users preference (each user prefers a different model). The second problem, on the other hand, can be treated more easily. This is because model risk can be quantified.

This thesis will cover two different problems in finance. Firstly, a fast and accurate pricing methodology for callable, i.e. early exercise, options will be developed. This methodology speeds up the quadrature pricing technique of Andricopoulos, Widdicks, Duck and Newton (2003), Andricopoulos, Widdicks, Duck and Newton (2004) and O'Sullivan (2004). Furthermore, we will extend the quadrature pricing technique to Heston's two dimensional stochastic volatility model. The development of these techniques is done with a view on quantifying model risk. These techniques can easily be used to price discrete barrier options and an adapted method is applicable for cliquets.

Secondly, model risk for pricing various exotic options will be considered. When talking about model risk, we have to distinguish between intra- and inter-model risk. We define intra-model risk for a certain contract as the maximum price difference within one model, given that the model is adequately calibrated to the initial market prices. These differences can arise due to different starting values for parameters or different objective functions in a calibration. Inter-model risk is the traditional model risk, i.e. the maximum price difference over various (ideally all) option pricing models, given that the models are adequately calibrated to initial market prices. Inter-model risk for equity derivatives is touched upon to some extent in, for example, Hull and Suo (2002) and Madan (2005), whereas intra-model risk is, as far as we are aware, only considered in Detlefsen and Härdle (2006). This document will cover both types of model risk.

For the pricing of exotic options several quantitative techniques are available. The pricing of purely callable exotics can be done with the quadrature technique as presented in this thesis. For the pricing of path-dependent options we will choose quadrature techniques and Monte Carlo based simulation techniques, depending on the model and the type of payoff.

The outline of this thesis is as follows. Chapter 1 discusses several models for pricing options. A fast and accurate pricing technique for callable options will be discussed in Chapter 2. This technique is compared to a technique of solving a Partial (Integro-) Differential Equation (P(I)DE) numerically and with the Monte Carlo simulation, treated in Chapters 3 and 4, respectively. Results and conclusions of this comparison can be found in Chapters 5 and 6. The exotics, for which the model risk will be determined, and their pricing methodologies are discussed in Chapter 7. Chapter 8 will give a short description of the calibration method used within this thesis. Results of all calibrated model parameters can be found in this chapter as well as the quantification of the model risk for the various exotic options.

0.1 Notation and Remarks

Throughout this thesis several variables and some loose notation is used. They are introduced in this section.

Variable	**Description**
$S(t)$	Price of the underlying asset at time t
S_0, S_T	Price of the underlying asset at time t_0 and T
K	The strike price
$x(t) = \log S(t)$	The logarithm of the asset price at time t
x_0, x_T	The logarithm of the asset price at time t_0 and T
t_0	Today's date
T	Maturity date
$\tau = T - t$	Time between t and T
r	Risk-free interest rate
$N(\cdot)$	The standard normal distribution function
$W(t)$	A Brownian motion
$\varphi = \begin{cases} -1 & \text{, for a put option} \\ 1 & \text{, for a call option} \end{cases}$	A useful parameter for defining puts and calls
$x^+ = \max(x, 0)$	The maximum between x and zero

Table 0.1.1: Notation

Some Remarks:

- The logarithm of the underlying asset price has proven to be more useful than the asset price itself. Therefore, most of the formulas are based on the logarithm of the asset price, $x(t)$.

- In this thesis expectations of some stochastic variable X are written as $\mathbb{E}_t[X]$, while this is actually the expectation under some risk neutral measure $\mathbb{Q}$ and given some filtration $\mathcal{F}_t$. So,
$$\mathbb{E}_t[X] = \mathbb{E}^{\mathbb{Q}}[X|\mathcal{F}_t].$$

- Many definitions of discrete Fourier transforms (DFT) exist. In this thesis the DFT $\widehat{\mathbf{f}} = \begin{bmatrix} \widehat{f}_1 & \cdots & \widehat{f}_N \end{bmatrix}^T$ of some vector $\mathbf{f} = \begin{bmatrix} f_1 & \cdots & f_N \end{bmatrix}^T$ is defined as

$$\widehat{f}_k = \sum_{n=1}^{N} f_n e^{-\frac{2\pi i}{N}(n-1)(k-1)}. \tag{0.1.1}$$

  The inverse transform then reads

$$f_k = \frac{1}{N} \sum_{n=1}^{N} \widehat{f}_n e^{\frac{2\pi i}{N}(n-1)(k-1)}. \tag{0.1.2}$$

Part I

Models

Chapter 1

Model Descriptions

This section presents the dynamics and some characteristics of the various models used in this thesis. Most of the characteristics are described, and not derived, because they already appear in the literature, see e.g. Cont and Tankov (2004), Dupire (1994) and Heston (1993).

1.1 Black-Scholes Model

The Black-Scholes model, introduced for option pricing, still forms the foundation for modern finance. Black and Scholes assume that the stock price follows a geometric Brownian motion. The dynamics of the logarithmic stock price are

$$dx\left(t\right)=\left(\mu-\frac{1}{2}\sigma^{2}\right)dt+\sigma dW\left(t\right), \tag{1.1.1}$$

where μ is the drift and σ the volatility. The price of a European option can be calculated by the Black-Scholes formula:

$$V=\varphi\left(S_{0}\Phi\left(\varphi d_{1}\right)-e^{-r(T-t_{0})}K\Phi\left(\varphi d_{2}\right)\right), \tag{1.1.2}$$

where the parameters d_1 and d_2 are defined as

$$\begin{aligned} d_{1} &= \frac{\log\left(\frac{S_{0}}{K}\right)+\left(r+\frac{1}{2}\sigma^{2}\right)\left(T-t_{0}\right)}{\sigma\sqrt{T-t_{0}}}, \\ d_{2} &= d_{1}-\sigma\sqrt{T-t_{0}} \end{aligned}$$

and Φ is the cumulative normal distribution function. For a complete derivation of this formula we refer to Black and Scholes (1973).

For the pricing of different kinds of options, e.g. American options, it may be useful to solve the relevant partial differential equation numerically. In the Black-Scholes world the option price V satisfies the following equation:

$$\frac{\partial V}{\partial\tau}=\frac{1}{2}\sigma^{2}\frac{\partial^{2}V}{\partial x^{2}}+\left(r-\frac{1}{2}\sigma^{2}\right)\frac{\partial V}{\partial x}-rV. \tag{1.1.3}$$

The initial and boundary conditions differ for different payoffs. A detailed derivation can be found in Wilmott (1998).

Finally, the characteristic function reads

$$\begin{aligned}\phi_T(\omega) &= \mathbb{E}_{t_0}\left[e^{i\omega x_T}\right] = \mathbb{E}_{t_0}\left[e^{i\omega\left(\left(r-\frac{1}{2}\sigma^2\right)T+\sigma W(T)\right)}\right] \\ &= e^{i\omega\left(r-\frac{1}{2}\sigma^2\right)T-\frac{1}{2}\sigma^2\omega^2 T}. \end{aligned} \tag{1.1.4}$$

1.2 Heston's Stochastic Volatility Model

Heston's stochastic volatility model (or Heston's model for short) allows the volatility to satisfy a SDE. The idea to model volatility as a random variable comes from market data that indicates the highly variable and unpredictable nature of volatility. Moreover, return distributions under stochastic volatility models typically have fatter tails than their lognormal counterparts, hereby being more realistic. The most cited argument to allow volatility to be random is that such models introduce a volatility smile. In other words, it captures the problem of the implied volatilities as described before to a large extent.

One of the most popular stochastic volatility models was introduced by Heston (1993). In Heston's model we consider two stochastic differential equations, one for the logarithmic asset price $x(t)$ and one for the (local) variance $v(t)$ of $x(t)$:

$$\begin{aligned} dx(t) &= \left(\mu - \frac{1}{2}v(t)\right)dt + \sqrt{v(t)}dW_1(t), \\ dv(t) &= -\lambda\left(v(t)-\overline{v}\right)dt + \eta\sqrt{v(t)}dW_2(t). \end{aligned} \tag{1.2.1}$$

Here $\lambda \geq 0$, $\overline{v} \geq 0$ and $\eta \geq 0$ are called the speed of mean reversion, the mean level of variance and the volatility of volatility, respectively. Furthermore, the Brownian motions $W_1(t)$ and $W_2(t)$ are assumed to be correlated with correlation coefficient ρ. The SDE for the variance can be recognized as a mean-reverting square root process, as originally proposed by Cox, Ingersoll and Ross (1985) to model the spot interest rate. One can easily deduce the name "mean-reverting": if $v(t)$ exceeds its mean $\overline{v}$, the term $-\lambda\left(v(t)-\overline{v}\right)dt$ drives the variance back to the mean. The same holds if $v(t)$ is below its mean.

The two dimensional pricing PDE for the Heston model can be deduced from hedging arguments. In case of a one-dimensional model a self-financing portfolio is constructed via an option and $-\Delta$ units of stocks. Whereas for Heston's two dimensional model the risk associated with the random volatility needs to be hedged as well. Hence, the self-financing portfolio Π consists of an option with value $V(S,v,t)$, $-\Delta$ units of the underlying asset and, in order to hedge the risk associated with the random volatility, $-\Delta_1$ units of another option with the same underlying and value $V_1(S,v,t)$:

$$\Pi = V - \Delta S - \Delta_1 V_1. \tag{1.2.2}$$

Using Itô's lemma and arbitrage arguments the following two dimensional pricing PDE is derived:

$$\frac{\partial V}{\partial \tau} = \frac{1}{2}v\frac{\partial^2 V}{\partial x^2} + \rho\eta v\frac{\partial^2 V}{\partial x \partial v} + \frac{1}{2}v\eta^2\frac{\partial^2 V}{\partial v^2} + \left(r - \frac{1}{2}v\right)\frac{\partial V}{\partial x} - \left(\lambda\left(v-\overline{v}\right) - \theta v\right)\frac{\partial V}{\partial v} - rV. \tag{1.2.3}$$

Here θ is known as the market price of risk. The market price of risk will be handled by the other parameters, because the model is calibrated to the market. So, θ can be chosen to equal zero:

$$\frac{\partial V}{\partial \tau} = \frac{1}{2}v\frac{\partial^2 V}{\partial x^2} + \rho\eta v\frac{\partial^2 V}{\partial x \partial v} + \frac{1}{2}v\eta^2\frac{\partial^2 V}{\partial v^2} + \left(r - \frac{1}{2}v\right)\frac{\partial V}{\partial x} - \lambda\left(v-\overline{v}\right)\frac{\partial V}{\partial v} - rV. \tag{1.2.4}$$

The characteristic function reads:

$$\phi_T(\omega) = e^{i\omega rT + \frac{v_0}{\eta^2}\left(\frac{1-e^{-DT}}{1-Ge^{-DT}}\right)(\lambda-\rho\eta i\omega-D) + \frac{\lambda\overline{v}}{\eta^2}\left(T(\lambda-\rho\eta i\omega-D) - 2\log\left(\frac{1-Ge^{-DT}}{1-G}\right)\right)}, \tag{1.2.5}$$

where

$$\begin{aligned} D &= \sqrt{(\lambda-\rho\eta i\omega)^2 + (\omega^2 + i\omega)\eta^2}, \\ G &= \frac{\lambda-\rho\eta i\omega-D}{\lambda-\rho\eta i\omega+D}. \end{aligned}$$

The characteristic function can be seen as an (imaginary) claim which must therefore satisfy (1.2.4). This is the key insight for the derivation of the characteristic function. For both (1.2.4) and (1.2.5) an elaborate derivation is presented in Heston (1993).

1.3 Bates Model

Bates (1996) extended the Heston model by considering jumps in the stock price process:

$$\begin{aligned} dx(t) &= \left(\mu - \frac{1}{2}v(t)\right)dt + \sqrt{v(t)}dW_1(t) + j_{N(t)}dN(t), \\ dv(t) &= -\lambda(v(t)-\overline{v})\,dt + \eta\sqrt{v(t)}dW_2(t). \end{aligned} \tag{1.3.1}$$

Here $N(t)$ is a Poisson process with intensity ξ and $j_{N(t)}$ are normally distributed jump sizes with expectation μ_j and variance σ_j^2, $j_{N(t)} \sim N\left(\mu_j, \sigma_j^2\right)$. The Poisson process $N(t)$ is independent of the Brownian motions and the jump sizes.

Under the martingale measure the characteristic function of Bates model reads, see e.g. Cont and Tankov (2004),

$$\begin{aligned} \phi_T(\omega) &= e^{i\omega(r-\kappa\xi)T + \frac{v_0}{\eta^2}\left(\frac{1-e^{-DT}}{1-Ge^{-DT}}\right)(\lambda-\rho\eta i\omega-D) + \frac{\lambda\overline{v}}{\eta^2}\left(T(\lambda-\rho\eta i\omega-D) - 2\log\left(\frac{1-Ge^{-DT}}{1-G}\right)\right)} \\ &\quad \times e^{\xi T\left(e^{i\omega\mu_j - \frac{1}{2}\sigma_j^2\omega^2} - 1\right)} \end{aligned} \tag{1.3.2}$$

where $\kappa = \mathbb{E}\left[e^{j_{N(t)}} - 1\right]$.

The Bates model has eight parameters while the Heston model has only five parameters for calibration. Because of these three extra parameters the Bates model can better fit the observed volatility surface, but stability over the parameters over time is more difficult to achieve.

1.4 Variance Gamma Model

The Variance Gamma (VG) process was first introduced in financial modelling by Madan and Seneta (1990) to cope with the shortcomings of the Black-Scholes model. The VG process is obtained by evaluating a drifted Brownian motion at random times given by a gamma process. The three parameters determining the VG process $X(t;\sigma,\nu,\theta)$ are: the volatility σ of the Brownian motion, the variance ν of the gamma distributed time and the drift θ of the time-changed Brownian motion with drift. The dynamics of the process are

$$dx(t) = -\widetilde{\alpha}dt + dX(t;\sigma,\nu,\theta), \tag{1.4.1}$$

where $-\widetilde{\alpha}$ is the drift of the logarithmic price of the asset and the VG process $X(t;\sigma,\nu,\theta)$ is defined as

$$X(t;\sigma,\nu,\theta) = \theta\gamma(t;1,\nu) + \sigma W(\gamma(t;1,\nu)). \tag{1.4.2}$$

Here $\gamma(t;1,\nu)$ denotes a gamma process with mean 1 and variance ν. The gamma process has the important property that increments are gamma distributed:

$$\gamma(t+s;1,\nu) - \gamma(s;1,\nu) \sim G(t,\nu).$$

From Cont and Tankov (2004) and Almendral and Oosterlee (2006) it is known that the pricing PIDE and the characteristic function are

$$\frac{\partial V}{\partial \tau} = r\frac{\partial V}{\partial x} - rV + \int_{-\infty}^{\infty}\left[V(x+y) - V(x) - (e^y - 1)\frac{\partial V}{\partial x}\right]k(y)\,dy \tag{1.4.3}$$

and

$$\phi_T(\omega) = e^{i\omega\left(r+\frac{1}{\nu}\log\left(1-\theta\nu-\frac{1}{2}\sigma^2\nu\right)\right)(T-t_0)-\frac{1}{\nu}\log\left(1-i\omega\theta\nu+\frac{1}{2}\omega^2\sigma^2\nu\right)(T-t_0)}, \tag{1.4.4}$$

respectively. In (1.4.3) $k(x)$ is known as the Lévy density and has the form

$$k(x) = \begin{cases} \frac{1}{\nu}\frac{e^{-\lambda_+|x|}}{|x|} & \text{if } x > 0 \\ \frac{1}{\nu}\frac{e^{-\lambda_-|x|}}{|x|} & \text{if } x < 0 \end{cases},$$

where the positive parameters $\lambda_\pm$ are given by

$$\lambda_\pm^{-1} = \sqrt{\frac{\theta^2\nu^2}{4} + \frac{\sigma^2\nu}{2}} \pm \frac{\theta\nu}{2}.$$

The parameters λ_+ and λ_- are the maximum and minimum allowed moments, respectively. So,

$$\mathbb{E}_{t_0}\left[S(t)^\lambda\right] < \infty, \text{ if and only if } \lambda_- < \lambda < \lambda_+.$$

For any sensible model we must at least have that the forward, $\mathbb{E}_{t_0}[S(t)]$, exists, therefore we must have $\lambda_+ > 1$.

1.5 The Local Volatility Model

The previous models discussed so far try to capture the flaws of the Black-Scholes model by adding a stochastic process, i.e. Heston assumes the volatility to be stochastic, Bates has a stochastic volatility and a stochastic jump process, and the VG model has a stochastic time. This section discusses a model, which was firstly presented in Dupire (1994), that deals with the volatility skew by introducing a deterministic volatility function. By this we mean that no extra random variables are involved. The dynamics of this model are

$$dx(t) = \left(\mu - \tfrac{1}{2}\sigma(x(t),t)^2\right)dt + \sigma(x(t),t)\,dW(t), \tag{1.5.1}$$

where $\sigma(x(t),t)$ is a deterministic function of the logarithm of the underlying and the time and is known as the local volatility function. The local volatility function equals

$$\sigma(x,t)^2 = 2\frac{\frac{\partial}{\partial T}C(x,x,t) + \mu\frac{\partial}{\partial k}C(x,x,t) + (r-\mu)C(x,x,t)}{\frac{\partial^2}{\partial k^2}C(x,x,t) - \frac{\partial}{\partial k}C(x,x,t)} \tag{1.5.2}$$

A detailed derivation is presented in Appendix A. In Dupire (1994) a local volatility function as a function of the spot price, instead of the logarithm of the spot price, is derived.

The pricing PDE of the local volatility model is equivalent to the Black-Scholes PDE and reads

$$\frac{\partial V}{\partial \tau} = \tfrac{1}{2}\sigma\left(x,\tau\right)^2 \frac{\partial^2 V}{\partial x^2} + \left(r - \tfrac{1}{2}\sigma\left(x,\tau\right)^2\right) \frac{\partial V}{\partial x} - rV. \tag{1.5.3}$$

Part II

Valuation Techniques

Chapter 2

Fourier Pricing Techniques

This section provides a detailed description of a novel technique for obtaining option prices based on Fourier transformations. Using Fourier transforms for the pricing of options is a relatively new approach in finance. Although it is already extensively described in literature, see e.g. Heston (1993), Carr and Madan (1999) and Bakshi and Madan (2000). The ability to solve options with early exercise opportunities is very recent. The idea for the algorithm described in this thesis originates from Lord (2005). Part of the research on this new approach is provided in this chapter. In section 2.1 we discuss the novel technique for one-dimensional models. This idea can be extended to multi-dimensional models. This is described for the Heston model in section 2.2. During the implementation several computational problems occurred. A discussion on the computational aspects can be found in section 2.3. Finally, an extrapolation technique to price American options in presented in section 2.4.

2.1 Pricing with Fourier Transforms for One-Dimensional Models

2.1.1 European Options

As discussed in Cox and Ross (1976), a perfect hedge can be formed by a European option and its underlying stock. The option can be valued by determining the discounted expected value of its payoff assuming risk neutrality. The European price is then

$$V(x,T) = e^{-r(T-t_0)}\mathbb{E}_{t_0}\left[\left(\varphi\left(e^{x_T}-K\right)\right)^+\right]. \tag{2.1.1}$$

This equation can be written as an integral:

$$V(x,T) = e^{-r(T-t_0)}\int_{-\infty}^{\infty} g(x_T)\, f(x_T|\, x)\, dx_T, \tag{2.1.2}$$

where $g(x_T) = \left(\varphi\left(e^{x_T}-K\right)\right)^+$ and $f(x_T|\, x)$ is the transition probability density from x at t_0 to x_T at T, such that $x_T - x$ can be interpreted as the logarithmic return from time t_0 to T. The models for which the routine as described here can be devised, satisfy

$$f(x_T|\, x) = f(z),$$

where $z = x_T - x$. Examples of such models are the one-dimensional exponential Lévy models. The key insight to fast pricing is the notion that, apart from the discounting, (2.1.2) represents

a correlation of g with the density function f. Premultiplying (2.1.2) by the factor $e^{\alpha x}$, ensuring that the Fourier transform exists, and taking its Fourier transform gives

$$\begin{aligned} e^{r(T-t_0)}\widehat{V}(\omega - i\alpha) &= e^{r(T-t_0)} \int_{-\infty}^{\infty} e^{i(\omega - i\alpha)x} V(x, t_0)\, dx \\ &= \int_{-\infty}^{\infty} e^{i(\omega - i\alpha)x} \int_{-\infty}^{\infty} g(x_T)\, f(x_T | x)\, dx_T dx \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{i(\omega - i\alpha)x} g(x_T)\, f(z)\, dz dx \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{i(\omega - i\alpha)x_T} g(x_T)\, e^{-i(\omega - i\alpha)z} f(z)\, dz dx_T \\ &= \int_{-\infty}^{\infty} e^{i(\omega - i\alpha)x_T} g(x_T)\, dx_T \int_{-\infty}^{\infty} e^{-i(\omega - i\alpha)z} f(z)\, dz \\ &= \widehat{g}(\omega - i\alpha)\, \widehat{f}(-(\omega - i\alpha)). \end{aligned} \tag{2.1.3}$$

Note that $\widehat{f}(\cdot)$ is the characteristic function, $\widehat{f}(\cdot) = \phi_T(\cdot)$ and $\widehat{g}(\cdot)$ is the Fourier transform of $g(x_T)$. The characteristic function is known explicitly. The other integral $\widehat{g}(\omega - i\alpha)$ can be calculated with any kind of numerical integration method, which is based on uniform grids. With non-uniform grid based numerical integration methods, e.g. Gaussian quadrature, we loose the applicability of the DFT. The computation of the continuous Fourier integral is described in detail is section 2.3.1. Finally, the option price $V(x, T)$ can be found by inverting the Fourier transformed option price:

$$V(x, T) = \frac{e^{-\alpha x} e^{-r(T-t_0)}}{2\pi} \int_{-\infty}^{\infty} e^{-i\omega x} \widehat{g}(\omega - i\alpha)\, \widehat{f}(-(\omega - i\alpha))\, d\omega. \tag{2.1.4}$$

Because the integrals are evaluated numerically we end up with a vector of option prices. The option price for some spot value $S_0 = e^{x_0}$ can be found by any interpolation method, e.g. by cubic splines, or by choosing the grid such that x_0 is a grid point.

2.1.2 Bermudan Options

A Bermudan option is an option which can be exercised on some prespecified exercise dates. Denote the set of exercise dates as $\mathcal{T} = \{t_1, ..., t_M\}$, $t_k < t_{k+1}$. With $g(x, t_k)$ the exercise payoff at $t_k \in \mathcal{T}$, just like the previous section, $C(x, t_k)$ the continuation value at this date and $V(x, t_k)$ the option value, we have, with $C(x, t_M) = 0$,

$$V(x, t_k) = \max(g(x, t_k), C(x, t_k)), \quad k = 1, ..., M, \tag{2.1.5}$$

$$C(x, t_k) = e^{-r(t_{k+1} - t_k)} \mathbb{E}[V(x, t_{k+1})], \quad k = 1, ..., M. \tag{2.1.6}$$

Notice that (2.1.6) can be approximated with the method described in the previous section. The price of the Bermudan is found by solving (2.1.5) and (2.1.6) M times.

An American option can be exercised at any time before maturity. In other words, it is a Bermudan option with an infinite amount of exercise dates. So, when choosing M very large an approximation for the American price is obtained. Alternatively, an extrapolation method exists to keep the number of time steps limited. This method is known as Richardson extrapolation and is discussed in section 2.4.

2.2 Pricing with Fourier Transforms for Heston's Model

For the Heston model the pricing with characteristic functions becomes more complicated than for one-dimensional models. In case of European options it can be treated similarly to the Black-Scholes model, however the pricing of callable options, e.g. Bermudans or Americans, is more difficult.

2.2.1 European Options

Recall that the price of a European option can be calculated as (see equation (2.1.1))

$$V(x_0, v_0, T) = e^{-r(T-t_0)} \mathbb{E}_{t_0} [g(x_T)], \tag{2.2.1}$$

where $g(x_T)$ presents the exercise payoff. The main difference between Heston's model and Black and Scholes' model in case of European options is that the transition density as described in section 2.1.1 is not only conditioned on the initial logarithmic stock price, but also on the initial variance. So, equation (2.2.1) in integral form reads

$$V(x_0, v_0, T) = e^{-r(T-t_0)} \int_{-\infty}^{\infty} g(x_T) f(x_T | x_0, v_0) \, dx_T. \tag{2.2.2}$$

Obviously, the same methodology as in section 2.1.1 applies to calculate expression (2.2.2).

2.2.2 Bermudan Options

The pricing of Bermudan options, or any other callable option, tends to be more complicated. This is because the exercise payoff at any time before maturity is dependent on the variance at that particular time. This section discusses this problem and presents a method to price Bermudans under the Heston model.

Now, consider a European style option with its payoff dependent on the underlying and the variance at maturity. With $g(x_T, v_T)$ the payoff function, the forward value of the option can be computed as

$$\begin{aligned} F(x_0, v_0) &= \mathbb{E}_{t_0} [g(x_T, v_T)] \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x_T, v_T) \, p(x_0 \to x_T, v_0 \to v_T) \, dx_T dv_T, \end{aligned} \tag{2.2.3}$$

with $p(x_0 \to x_T, v_0 \to v_T)$ being the transition probability density from x_0 and v_0 at t_0 to x_T and v_T at T, and where x_0, v_0, x_T and v_T represent the logarithm of the initial stock price, initial variance, the logarithm of the stock price at maturity and the variance at maturity, respectively. In the two-dimensional case the transition density cannot be written in terms of the changes in the logarithmic stock price and the variance. So, the identity $p(x_0 \to x_T, v_0 \to v_T) = p(z_x, z_v)$, with $z_x = x_T - x_0$ and $z_v = v_T - v_0$, does not hold in general. Instead, we use the following lemma.

Lemma 1 *The transition density for Heston's model (1.2.1) can be written in terms of the change in the logarithmic stock price and the variance at maturity, given today's variance:*

$$p(x_0 \to x_T, v_0 \to v_T) = p(z_x, v_T | v_0), \tag{2.2.4}$$

where $z_x = x_T - x_0$.

Proof. We use the Gil-Pelaez inversion formula on the conditional characteristic function. For the derivation of the two-dimensional conditional characteristic function of the Heston model, we need to refer to advanced results from Duffie, Pan and Singleton (2000), which indicates that for affine diffusion processes the characteristic function of x_T equals

$$\mathbb{E}_{t_0}\left[e^{i\omega x_T + i\xi v_T}\right] = e^{A(\omega,\xi,\tau)+B(\omega,\xi,\tau)x_{t_0}+C(\omega,\xi,\tau)v_{t_0}}, \tag{2.2.5}$$

where $\tau = T - t_0$ and A, B, C constants to be determined. Obviously, at $\tau = 0$ we have

$$\mathbb{E}_T\left[e^{i\omega x_T + i\xi v_T}\right] = e^{i\omega x_T + i\xi v_T} = e^{A(\omega,\xi,0)+B(\omega,\xi,0)x_T+C(\omega,\xi,0)v_T}, \tag{2.2.6}$$

leading to the following boundary conditions:

$$A(\omega,\xi,0) = 0, \tag{2.2.7}$$
$$B(\omega,\xi,0) = i\omega \tag{2.2.8}$$

and

$$C(\omega,\xi,0) = i\xi. \tag{2.2.9}$$

Since $x(t)$ does not appear in the SDE for $v(t)$, in neither the drift nor the volatility, it can be shown that $B(\omega,\xi,\tau) = i\omega$. Using the Gil-Pelaez inversion formula, we then find the following expression for the transition density:

$$\begin{aligned} p(x_0 \to x_T, v_0 \to v_T) &= \frac{1}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathrm{Re}\left(e^{A(\omega,\xi,\tau)+i\omega x_{t_0}+C(\omega,\xi,\tau)v_{t_0}}e^{-i\omega x_T - i\xi v_T}\right) d\omega d\xi \\ &= \frac{1}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathrm{Re}\left(e^{A(\omega,\xi,\tau)-i\omega z_x+C(\omega,\xi,\tau)v_{t_0}-i\xi v_T}\right) d\omega d\xi, \end{aligned}$$

which clearly depends on z_x, v_{t_0} and v_T. The fact that the characteristic function is conditioned on time t_0 completes the proof. ■

With Lemma 1, (2.2.3) can now be written as

$$F(x_0,v_0) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x_T,v_T)\, p(z_x,\, v_T|\, v_0)\, dz_x dv_T, \tag{2.2.10}$$

which equals

$$F(x_0,v_0) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x_T,v_T)\, p(\, z_x|\, v_T,v_0)\, p_v(\, v_T|\, v_0)\, dz_x dv_T. \tag{2.2.11}$$

Here $p_v(\, v_T|\, v_0)$ is the transition density of the variance. Feller (1951) has shown that the density of v_T conditioned on v_0 reads

$$p_v(\, v_T|\, v_0) = ce^{-b-x}\left(\frac{x}{b}\right)^{\frac{(a-1)}{2}} I_{a-1}\left(2\sqrt{bx}\right), \tag{2.2.12}$$

where $c = 2\lambda/\left(\left(1-e^{-\lambda T}\right)\eta^2\right)$, $b = cv_0e^{-\lambda T}$, $x = cv_T$ and $a = 2\lambda\overline{v}/\eta^2$. Here $I_a(x)$ is the modified Bessel function of the first kind, see e.g. Abramowitz and Stegun (1972). Premultiplying

(2.2.11) by $e^{\alpha x_0}$, and taking the Fourier transform, we find

$$\begin{aligned}
\widehat{F}(\omega - i\alpha, v_0) &= \int_{-\infty}^{\infty} e^{i(\omega - i\alpha)x_0} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x_T, v_T)\, p(z_x|\, v_T, v_0)\, p_v(v_T|\, v_0)\, dz_x dv_T dx_0 \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{i(\omega - i\alpha)x_0} g(x_T, v_T)\, p(z_x|\, v_T, v_0)\, p_v(v_T|\, v_0)\, dz_x dv_T dx_0 \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{i(\omega - i\alpha)x_T} g(x_T, v_T) \\
&\qquad e^{-i(\omega - i\alpha)z_x} p(z_x|\, v_T, v_0)\, p_v(v_T|\, v_0)\, dz_x dv_T dx_T \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{i(\omega - i\alpha)x_T} g(x_T, v_T)\, dx_T \\
&\qquad \cdot \int_{-\infty}^{\infty} e^{-i(\omega - i\alpha)z_x} p(z_x|\, v_T, v_0)\, dz_x \cdot p_v(v_T|\, v_0)\, dv_T \\
&= \int_{-\infty}^{\infty} \widehat{g}(\omega - i\alpha, v_T) \cdot \widehat{p}(-(\omega - i\alpha)) \cdot p_v(v_T|\, v_0)\, dv_T, \qquad (2.2.13)
\end{aligned}$$

where $\widehat{p}(\omega)$ is the characteristic function of the logarithm of the stock given v_0 and given v_T, which is known analytically:

$$\widehat{p}(\omega) = e^{i\omega\left[r(T-t_0) + \frac{\rho}{\eta}(v_T - v_0 - \lambda\overline{v}(T-t_0))\right]} \Phi\left(\omega\left(\frac{\lambda\rho}{\eta} - \frac{1}{2}\right) + \frac{1}{2}i\omega^2\left(1-\rho^2\right)\right), \qquad (2.2.14)$$

with $\Phi(a)$ the characteristic function of $\int_{t_0}^{T} v(s)\, ds$ given v_0 and given v_T. In Broadie and Kaya (2004) it has been shown that

$$\begin{aligned}
\Phi(a) &= \frac{\gamma(a)\, e^{-\frac{1}{2}(\gamma(a)-\lambda)(T-t_0)}\left(1 - e^{-\lambda(T-t_0)}\right)}{\lambda\left(1 - e^{-\gamma(a)(T-t_0)}\right)} \\
&\quad \times e^{\frac{v_0+v_T}{\eta^2}\left[\frac{\lambda\left(1+e^{-\lambda(T-t_0)}\right)}{1-e^{-\lambda(T-t_0)}} - \frac{\gamma(a)\left(1+e^{-\gamma(a)(T-t_0)}\right)}{1-e^{-\gamma(a)(T-t_0)}}\right]} \\
&\quad \times \frac{I_{\frac{1}{2}d-1}\left[\sqrt{v_0 v_T}\frac{4\gamma(a)e^{-\frac{1}{2}\gamma(a)(T-t_0)}}{\eta^2\left(1-e^{-\gamma(a)(T-t_0)}\right)}\right]}{I_{\frac{1}{2}d-1}\left[\sqrt{v_0 v_T}\frac{4\lambda e^{-\frac{1}{2}\lambda(T-t_0)}}{\eta^2\left(1-e^{-\lambda(T-t_0)}\right)}\right]}, \qquad (2.2.15)
\end{aligned}$$

where $\gamma(a) = \sqrt{\lambda^2 - 2\eta^2 ia}$, $d = 4\overline{v}\lambda/\eta^2$ and $I_\nu(x)$ is the modified Bessel function of the first kind. A derivation of (2.2.14) is presented in Appendix B.

Notice that for a Bermudan option, we need to define a grid in both the log-stock direction and the variance direction (as both are not known for future exercise dates). In other words, for every time step and "initial" variance we need to calculate (2.2.13) to determine the price of a Bermudan. The integral in (2.2.13) can be calculated by any numerical integration method, e.g. by the trapezoidal rule. Denote the number of grid points in log-stock direction by N and the number of grid points in variance direction by J, then the computational complexity of the Fourier transform $\widehat{g}(\omega - i\alpha, v_T)$ is $\mathcal{O}(N \log N)$. The numerical integration method is $\mathcal{O}(J)$. So, the complete computational cost of equation (2.2.13) is $\mathcal{O}(JN \log N)$. This computation needs to be done for each exercise date and "initial" variance, resulting in a total computational cost

for a Bermudan of $\mathcal{O}\left(MJ^2N\log N\right)$. The number of grid points in variance direction, J, may be kept small by using Gaussian quadrature and non-linear grids. More on this topic can be found in section (2.3.4).

2.3 Computational Aspects

This section details on some computational aspects of the implementation of the methodology described in the previous sections. Firstly, the computation of the continuous Fourier integral will be treated. Then we discuss the computation of a complex logarithm and we end with a suggestion on how to compute integral (2.2.13) efficiently.

2.3.1 Computing the Continuous Fourier Transform

This section discusses two numerical techniques to approximate the continuous Fourier transform. Both are based on uniform grids and use the FFT algorithm to increase the efficiency. Recall that the Fourier transform of $h(x)$ is defined as

$$\widehat{h}(\omega) = \int_{-\infty}^{\infty} e^{i\omega x} h(x)\, dx \tag{2.3.1}$$

Newton-Cotes

The Newton-Cotes method is a collective name for a number of quadrature methods, e.g. the trapezoidal rule or Simpson's rule. In a general form the Newton-Cotes method for the continuous Fourier integral can be written as

$$\widehat{h}(\omega) = \sum_{n=1}^{N} e^{i\omega x_n} h(x_n)\, b_n \Delta x + \mathcal{O}\left(h^j\right), \tag{2.3.2}$$

were $x_n = L + (n-1)\,\Delta x$ for $n = 1, .., N$, $\Delta x = \frac{U-L}{N-1}$ and b_n is used for the boundary correction. Note that with these settings the integral is truncated at L and U. Both b_n and j depend on the type of Newton-Cotes method, e.g. for the trapezoidal rule $b_n = \begin{cases} \frac{1}{2} & \text{, for } n = 1, N \\ 1 & \text{, otherwise} \end{cases}$ and $j = 2$. In order to apply the DFT, ω needs to be discretized conform the following equality:

$$\omega_m = \omega_1 + (m-1)\,\Delta\omega \text{ for } m = 1, ..., N, \tag{2.3.3}$$

where $\Delta\omega = \frac{2\pi}{N\Delta x}$. This gives

$$\begin{aligned}
\widehat{h}(\omega_m) &= \sum_{n=1}^{N} e^{i\omega_m x_n} h(x_n)\, b_n \Delta x + \mathcal{O}\left(h^j\right) \\
&= \sum_{n=1}^{N} e^{i(\omega_1 + (m-1)\Delta\omega)(L+(n-1)\Delta x)} h(x_n)\, b_n \Delta x + \mathcal{O}\left(h^j\right) \\
&= \sum_{n=1}^{N} e^{i(m-1)(n-1)\frac{2\pi}{N} + L(m-1)\Delta\omega + \omega_1 x_n} h(x_n)\, b_n \Delta x + \mathcal{O}\left(h^j\right) \\
&= e^{L(m-1)\Delta\omega} \Delta x \sum_{n=1}^{N} e^{i(m-1)(n-1)\frac{2\pi}{N}} e^{\omega_1 x_n} h(x_n)\, b_n + \mathcal{O}\left(h^j\right).
\end{aligned} \tag{2.3.4}$$

Usually the grid is chosen to be symmetric and so $\omega_1 = -\frac{N-1}{2}\Delta\omega$. Notice that the first part of the sum looks like the inverse DFT as defined in (0.1.2), which can be computed by the FFT. So,

$$\widehat{h}(\omega_m) = e^{L(m-1)\Delta\omega}\Delta x \cdot N \cdot \text{ifft}\left(e^{\omega_1 \mathbf{x}} h(\mathbf{x})\,\mathbf{b}\right)_m + \mathcal{O}\left(h^j\right). \tag{2.3.5}$$

Here we define

$$e^{\omega_1 \mathbf{x}} = \begin{bmatrix} e^{\omega_1 x_1} \\ \vdots \\ e^{\omega_1 x_N} \end{bmatrix}, \; h(\mathbf{x}) = \begin{bmatrix} h(x_1) \\ \vdots \\ h(x_N) \end{bmatrix}, \; \mathbf{b} = \begin{bmatrix} b_1 \\ \vdots \\ b_N \end{bmatrix},$$

$\text{ifft}(\cdot)_m$ is the m-th element of the inverse Fourier transform and all vector multiplications are pointwise.

Kernel Functions

Not uncommonly, the calculation of accurate numerical values for the Fourier integral is quite difficult. The reason for this is that these integrands may be highly oscillatory as pointed out in the following example.

Example 1 *Consider the integrand of (2.3.1), where $h(x)$ is the standard normal probability density. The real part of this integrand has the form $\cos(\omega x)\,h(x)$ and the imaginary part $\sin(\omega x)\,h(x)$. In general $h(x)$ is not oscillatory, which results in an oscillatory nature for both the real and imaginary part. This can be seen in the following plot:*

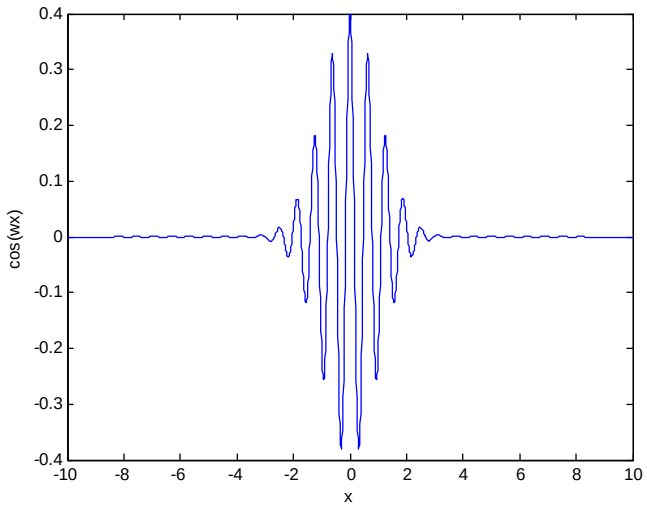

Figure 2.3.1: The oscillatory nature of the Fourier integrand

Approximating such an integral by any kind of Newton-Cotes formula may be inaccurate, because Newton-Cotes formulas only evaluate the integrand at fixed grid points. Increasing the number of grid points results in a finer mesh in the real domain, however in Fourier space the step size remains equal and the grid size increases. In other words, the integrand needs to be evaluated for higher values of ω, which gives rise to an even more oscillatory integrand. Here

we will present an accurate method, which overcomes this problem (Press, Teukolsky, Vetterling and Flannery, 1992).

Let us first truncate the integral at L and U. Divide the interval $[L,U]$ into $(N-1)$ subintervals, where N is the number of grid points, and define

$$\Delta x = \frac{U-L}{N-1},\ x_n = L + (n-1)\,\Delta x,\ h_n = h\,(x_n) \text{ and } n = 1, ..., N.$$

Given the sample points h_n we can approximate the function $h\,(x)$ everywhere in the interval $[L,U]$ by interpolation from nearby points n, e.g. by linear interpolation. The formulas for such interpolation schemes are (piecewise) polynomials in the independent variable x, but with coefficients that are linear in the function values h_n. Interpolation can be viewed as a technique to approximate a function by a sum of kernel functions times sample values. Let us write

$$h\,(x) \approx \sum_{n=1}^{N} h_n \psi\left(\frac{x-x_n}{\Delta x}\right) + \sum_{n=endpoints} h_n \varphi_n\left(\frac{x-x_n}{\Delta x}\right), \tag{2.3.6}$$

where for linear interpolation we have

$$\psi\,(s) = \begin{cases} 1+s & \text{, for } -1 \le s \le 0 \\ 1-s & \text{, for } 0 < s \le 1 \\ 0 & \text{, otherwise} \end{cases}$$

and

$$\varphi_1\,(s) = \begin{cases} -1-s & \text{, for } -1 \le s < 0 \\ 0 & \text{, otherwise} \end{cases},\ \varphi_N\,(s) = \begin{cases} s-1 & \text{, for } 0 < s \le 1 \\ 0 & \text{, otherwise} \end{cases}.$$

Notice that in the first sum of (2.3.6) we have included all the points, so the φ_n's are actually differences between true endpoint kernels and the interior kernel ψ. Application of the Fourier transformation to (2.3.6) gives

$$\begin{aligned} \int_L^U e^{i\omega x} h\,(x)\,dx &\approx \int_L^U e^{i\omega x} \sum_{n=1}^{N} h_n \psi\left(\frac{x-x_n}{\Delta x}\right) dx + \int_L^U e^{i\omega x} \sum_{n=endpoints} h_n \varphi_n\left(\frac{x-x_n}{\Delta x}\right) dx \\ &\approx \sum_{n=1}^{N} h_n \int_L^U e^{i\omega x} \psi\left(\frac{x-x_n}{\Delta x}\right) dx + \sum_{n=endpoints} h_n \int_L^U e^{i\omega x} \varphi_n\left(\frac{x-x_n}{\Delta x}\right) dx \\ &\approx \Delta x e^{i\omega L}\left[W\,(\omega\Delta x) \sum_{n=1}^{N} e^{i(n-1)\omega\Delta x} h_n + \sum_{n=endpoints} h_n \alpha_n\,(\omega\Delta x)\right]. \end{aligned} \tag{2.3.7}$$

Here $W\,(\omega\Delta x)$ and $\alpha_n\,(\omega\Delta x)$ are defined by

$$W\,(\omega\Delta x) = \int_{-\infty}^{\infty} e^{i\omega\Delta x s} \psi\,(s)\,ds, \tag{2.3.8}$$

$$\alpha_n\,(\omega\Delta x) = \int_{-\infty}^{\infty} e^{i\omega\Delta x s} \varphi_j\,(s-n+1)\,ds. \tag{2.3.9}$$

The key point is that (2.3.8) and (2.3.9) can be evaluated analytically for any given interpolation scheme. For the linear interpolation scheme we have

$$\begin{aligned} W(\omega\Delta x) &= \frac{2(1-\cos(\omega\Delta x))}{(\omega\Delta x)^2}, \\ \alpha_1(\omega\Delta x) &= \frac{i(\omega\Delta x-\sin(\omega\Delta x))+\cos(\omega\Delta x)-1}{(\omega\Delta x)^2}, \\ \alpha_N(\omega\Delta x) &= e^{2i\omega x^*}\overline{\alpha_1}(\omega\Delta x), \end{aligned}$$

where $\overline{\alpha_0}$ is the complex conjugate of α_0. It is any easy exercise to find these results. For higher order interpolation schemes we refer to Press et al. (1992).

Now, the sum $\sum_{n=1}^{N} e^{i(n-1)\omega\Delta x}h_n$ remains to be calculated. For the discrete Fourier transform to exist, we satisfy

$$\Delta\omega = \frac{2\pi}{N\Delta x},$$

and we define a grid as follows:

$$\omega_m = \omega_1 + (m-1)\Delta\omega, \text{ for } m = 1, ..., N$$

This gives

$$\sum_{n=1}^{N} e^{i(n-1)\omega\Delta x}h_n = \sum_{n=1}^{N} e^{i(n-1)(m-1)\frac{2\pi}{N}}e^{i(n-1)\omega_1}h_n. \tag{2.3.10}$$

Notice the similarity to the definition of the inverse discrete Fourier transform (0.1.2). Equation (2.3.10) can therefore be evaluated efficiently by the FFT algorithm. For linear interpolation we can now write (2.3.7) in its final form as

$$\widehat{h}(\omega_m)\,dx \approx \Delta x e^{i\omega_m L}\left\{\mathbf{W}\cdot N\cdot ifft\left(h_1\ e^{i\omega_1}h_2\ ...\ e^{i(N-1)\omega_1}h_N\right)_m + h_1\boldsymbol{\alpha}_1 + h_N\boldsymbol{\alpha}_N\right\},$$

where $\mathbf{W} = [W(\omega_1\Delta x)\ ...\ W(\omega_N\Delta x)]^T$ and $\boldsymbol{\alpha}_n = [\alpha_n(\omega_1\Delta x)\ ...\ \alpha_n(\omega_N\Delta x)]^T$ for $n = 1, N$.

2.3.2 The Logarithm of a Complex Variable

The calculation of the characteristic function for Heston's model includes the calculation of the logarithm of a complex variable. Let us first consider the characteristic function conditioned on v_0 (for European options):

$$\begin{aligned} \phi_T(\omega) = \exp\Bigg(&i\omega rT + \frac{v_0}{\eta^2}\left(\frac{1-e^{-DT}}{1-Ge^{-DT}}\right)(\lambda-\rho\eta i\omega-D) \\ &+\frac{\lambda\overline{v}}{\eta^2}\left(T(\lambda-\rho\eta i\omega-D)-2\log\left(\frac{1-Ge^{-DT}}{1-G}\right)\right)\Bigg), \end{aligned} \tag{2.3.11}$$

where

$$\begin{aligned} D &= \sqrt{(\lambda-\rho\eta i\omega)^2+(\omega^2+i\omega)\eta^2}, \\ G &= \frac{\lambda-\rho\eta i\omega-D}{\lambda-\rho\eta i\omega+D}. \end{aligned}$$

Notice that (2.3.11) contains the logarithm of a complex-valued variable:

$$\log\left(\frac{1-Ge^{-DT}}{1-G}\right).$$

The logarithm of a complex number is a multivalued function. Consider a complex variable $z = a + ib = re^{i(t+2\pi n)}$ with $t \in (-\pi, \pi]$ the principal argument and $n \in \mathbb{Z}$, then the logarithm of a complex variable equals

$$\log(z) = \log|r| + i(t + 2\pi n). \tag{2.3.12}$$

A problem may occur during the implementation as most programming languages assume that $(t + 2\pi n) \in (-\pi, \pi]$. This will give rise to discontinuities in the complex logarithm, shown in the following example.

Example 2 *Consider a complex variable $z = a + ib = re^{i\varphi}$. For $\varphi = \frac{3}{2}\pi$, Matlab considers $n = -1$ in (2.3.12) and calculates*

$$\log(z) = \log|r| - \frac{1}{2}\pi i,$$

whereas it actually should equal to

$$\log(z) = \log|r| + \frac{3}{2}\pi i.$$

If we look at the graph of the imaginary part of $\log(z)$, for $\varphi \in [0, 10\pi]$, we notice the following difference between the red line, the correct solution, and the blue (zigzagged) line, i.e. the solution of Matlab.

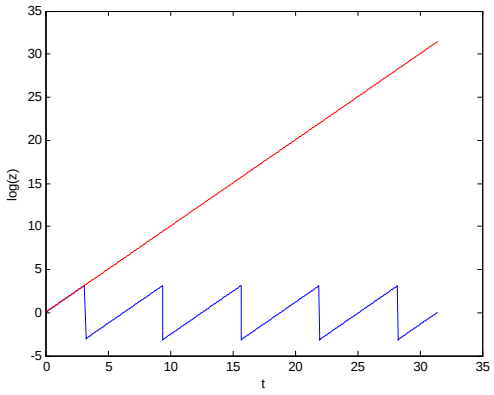

Figure 2.3.2: The logarithm of a complex variable

Lord and Kahl (2006) have shown that in formulation (2.3.11) the logarithm remains in its principal branch. In other words, we do not need to bother about it. In the Bermudan case, however, we do need to bother about it. Consider the characteristic function conditioned on v_0

and v_T (B.0.6). This function requires the calculation of modified Bessel functions of the first kind. The modified Bessel function of the first kind is defined by

$$I_\nu(z) = \left(\frac{1}{2}z\right)^\nu \sum_{k=0}^{\infty} \frac{\left(\frac{1}{4}z^2\right)^k}{k!\Gamma(\nu+k+1)}, \tag{2.3.13}$$

where $\Gamma(a)$ is the gamma function. If z is complex-valued it is necessary to calculate the logarithm of a complex variable, because

$$\left(\frac{1}{2}z\right)^\nu = \left(\frac{1}{2}\right)^\nu z^\nu = \left(\frac{1}{2}\right)^\nu e^{\nu \log z}.$$

Recall that in the case of (B.0.6), we have

$$z = \sqrt{v_0 v_T}\frac{4\gamma(a)\, e^{-\frac{1}{2}\gamma(a)(T-t_0)}}{\eta^2\left(1-e^{-\gamma(a)(T-t_0)}\right)}.$$

From Lord and Kahl (2006) we know that if we evaluate $f(a) := \frac{\gamma(a)e^{-\frac{1}{2}\gamma(a)(T-t_0)}}{\left(1-e^{-\gamma(a)(T-t_0)}\right)}$ as

$$\ln(f(a)) = g(a) - \mathrm{Im}(g(i\cdot \mathrm{Im}(a))) + \mathrm{Im}(\ln(f(i\cdot \mathrm{Im}(a))))$$

where

$$g(a) = \ln(\gamma(a)) - \frac{1}{2}\gamma(a)(T-t_0) - \ln\left(1-e^{-\gamma(a)(T-t_0)}\right),$$

then the logarithm is kept continuous. Here the complex logarithm is restricted to its principal branch. For details we refer to Lord and Kahl (2006).

2.3.3 The Approximation of Almost Zero

In order to apply the fast Fourier Transform algorithm for the computation of a continuous Fourier transform it is required that the step size in Fourier domain is constrained by

$$\Delta\omega = \frac{2\pi}{N\Delta x}.$$

Here $\Delta x = \frac{U-L}{N-1}$, which gives

$$\Delta\omega = \frac{2\pi(N-1)}{N(U-L)}.$$

Obviously, the step size in Fourier space is increasing with the number of grid points. Therefore, when the number of grid points is increasing the size of the grid in Fourier space is increasing too. As a result, values of ω can get very large. Now, consider the characteristic function of $\int_{t_0}^{T} v(s)\,ds$ given v_0 and given v_T:

$$\begin{aligned}\Phi(a) \quad = \quad & \frac{\gamma(a)\, e^{-\frac{1}{2}(\gamma(a)-\lambda)(T-t_0)}\left(1-e^{-\lambda(T-t_0)}\right)}{\lambda\left(1-e^{-\gamma(a)(T-t_0)}\right)} \\ & \times e^{\frac{v_0+v_T}{\eta^2}\left[\frac{\lambda\left(1+e^{-\lambda(T-t_0)}\right)}{1-e^{-\lambda(T-t_0)}} - \frac{\gamma(a)\left(1+e^{-\gamma(a)(T-t_0)}\right)}{1-e^{-\theta(a)(T-t_0)}}\right]} \\ & \times \frac{I_{\frac{1}{2}d-1}\left[\sqrt{v_0 v_T}\frac{4\gamma(a)e^{-\frac{1}{2}\gamma(a)(T-t_0)}}{\eta^2\left(1-e^{-\gamma(a)(T-t_0)}\right)}\right]}{I_{\frac{1}{2}d-1}\left[\sqrt{v_0 v_T}\frac{4\lambda e^{-\frac{1}{2}\lambda(T-t_0)}}{\eta^2\left(1-e^{-\lambda(T-t_0)}\right)}\right]},\end{aligned}$$

where $\gamma(a) = \sqrt{\lambda^2 - 2\eta^2 ia}$, $a = \omega\left(\frac{\lambda\rho}{\eta} - \frac{1}{2}\right) + \frac{1}{2}i\omega^2\left(1-\rho^2\right)$ and $d = 4\overline{v}\lambda/\eta^2$. Notice that a and $\gamma(a)$ become large for large ω, resulting in a value $z = \sqrt{v_0 v_T}\frac{4\gamma(a)e^{-\frac{1}{2}\gamma(a)(T-t_0)}}{\eta^2\left(1-e^{-\gamma(a)(T-t_0)}\right)}$ which is approximately zero. In most of the programming languages z will be smaller than the smallest positive value and it will be set equal to zero. On the other hand, $\nu = \frac{1}{2}d - 1$ will most likely be negative and during the computation of the modified Bessel function the quantity $z^\nu = r^\nu e^{\nu it}$ needs to be calculated. As ν is negative and r is set to zero, r^ν is undefined. Whenever this occurs r should be approximated by the smallest positive value in a programming language, e.g. in Matlab this value is known to be *"realmin"*.

2.3.4 Computing the Integral

As stated, the pricing of Bermudan options is of $\mathcal{O}\left(MJ^2N\log N\right)$, where M is the number of exercise dates, J the number of grid points in variance direction and N the number of grid points in the log-spot direction. Obviously, J contributes significantly to the total computational costs, therefore we wish to keep this parameter as small as possible. In order to remain a high accuracy the integral must be computed efficiently.

Firstly, it is advisable to use Gaussian quadrature rules to evaluate the integral numerically. With a limited number of grid points it is more accurate than for example the trapezoidal rule or Simpson's rule. The Gaussian quadrature is described in Burden and Faires (2001). Secondly, more accuracy is gained by stretching the grid. Consider that the following integral needs to be computed numerically:

$$I = \int_a^b f(x)\,dx. \tag{2.3.14}$$

A grid stretching function $g(x)$ gives the stretched grid: $y = g(x)$. Applying this stretching function gives

$$I = \int_{g(a)}^{g(b)} f\left(g^{-1}(y)\right)\frac{1}{g'\left(g^{-1}(y)\right)}dy. \tag{2.3.15}$$

Numerical testing showed that $g(x) = x^{\frac{1}{5}}$ provides good results for equation (2.2.13), in which case I equals

$$I = 5\int_0^{b^{\frac{1}{5}}} f\left(y^5\right)y^4 dy. \tag{2.3.16}$$

2.4 American Options with Richardson Extrapolation

Numerically, the value of an American option can be approximated either by a Bermudan option with many early exercise dates or by Richardson extrapolation on a series of Bermudan options with increasing early exercise dates. In an important contribution, Geske and Johnson (1984) showed that it was possible to value an American style option by using a series of Bermudan options. Chang, Chung and Stapleton (2002) improved their ideas by introducing the repeated Richardson extrapolation algorithm, which will be explained in this section.

Richardson extrapolation is a technique to generate accurate results within a limited amount of time. It is especially applicable to situations where the error depends on a parameter, e.g. the time step Δt. To illustrate the procedure assume we have an approximation $q(\Delta t)$ to some quantity Q. Furthermore, assume that the error of the approximation is $\mathcal{O}(\Delta t)$ and smooth, then Q is expended as follows:

$$Q = q(\Delta t) + K_1 \Delta t + K_2 \Delta t^2 + K_3 \Delta t^3 + \cdots \tag{2.4.1}$$

where the K_i's are known constants. Now, if we halve the time step to $\frac{\Delta t}{2}$ we will obtain a twice as accurate result:

$$Q = q\left(\frac{\Delta t}{2}\right) + K_1 \frac{\Delta t}{2} + K_2 \frac{\Delta t^2}{4} + K_3 \frac{\Delta t^3}{8} + \cdots . \tag{2.4.2}$$

Richardson extrapolation combines (2.4.1) and (2.4.2) to obtain a higher order of accuracy. By multiplying (2.4.2) by 2 and subtracting (2.4.1) one obtains

$$Q = q\left(\frac{\Delta t}{2}\right) + \left[q\left(\frac{\Delta t}{2}\right) - q(\Delta t)\right] + K_2\left(\frac{\Delta t^2}{2} - \Delta t^2\right) + K_3\left(\frac{\Delta t^3}{4} - \Delta t^3\right) + \cdots . \tag{2.4.3}$$

In this way, the new approximation, $q\left(\frac{\Delta t}{2}\right) + \left[q\left(\frac{\Delta t}{2}\right) - q(\Delta t)\right]$, is $\mathcal{O}\left(\Delta t^2\right)$ accurate. This idea is repeated in the repeated Richardson extrapolation algorithm. Let us denote

$$q_1(\Delta t) = q(\Delta t) \tag{2.4.4}$$

and

$$q_2(\Delta t) = q_1\left(\frac{\Delta t}{2}\right) + \left[q_1\left(\frac{\Delta t}{2}\right) - q_1(\Delta t)\right]. \tag{2.4.5}$$

Then we have

$$Q = q_2(\Delta t) - \frac{K_2}{2}\Delta t^2 - \frac{3K_3}{4}\Delta t^3 + \cdots . \tag{2.4.6}$$

Halving the step size gives

$$Q = q_2\left(\frac{\Delta t}{2}\right) - \frac{K_2}{8}\Delta t^2 - \frac{3K_3}{32}\Delta t^3 + \cdots . \tag{2.4.7}$$

Subtracting (2.4.6) from 4 times (2.4.7) the $\mathcal{O}\left(\Delta t^2\right)$ term is eliminated:

$$Q = q_2\left(\frac{\Delta t}{2}\right) + \frac{q_2\left(\frac{\Delta t}{2}\right) - q_2(\Delta t)}{3} + \frac{3K_3}{8}\Delta t^3 + \cdots . \tag{2.4.8}$$

Now, if we denote $q_3(\Delta t) = q_2\left(\frac{\Delta t}{2}\right) + \frac{q_2\left(\frac{\Delta t}{2}\right) - q_2(\Delta t)}{3}$ we could apply the same idea and so on. In summary, the repeated Richardson extrapolation reads

$$q_j(\Delta t) = q_{j-1}\left(\frac{\Delta t}{2}\right) + \frac{q_{j-1}\left(\frac{\Delta t}{2}\right) - q_{j-1}(\Delta t)}{2^{j-1} - 1} \tag{2.4.9}$$

and the error is

$$Q - q_j(\Delta t) = \mathcal{O}\left(\Delta t^j\right). \tag{2.4.10}$$

The valuation of American options by Richardson extrapolation on a series of Bermudan options with increasing early exercise dates is done with the following algorithm:

Algorithm 1 *Denote $P(M)$ as the price of a Bermudan option with M exercise dates, then for $j = 1, 2, 3, ..., k$, set $A_{j,0} = P\left(\widetilde{M} \cdot 2^{j-1}\right)$, and compute for $m = 1, 2, ..., k-1$:*

$$A_{j,m} = A_{j+1,m-1} + \frac{A_{j+1,m-1} - A_{j,m-1}}{2^m - 1} \tag{2.4.11}$$

Example 2 *In this example the algorithm is shown for $k = 3$ and $\widetilde{M} = 1$ in Table 2.4.1.*

$A_{1,0} = P(1)$	$A_{1,1} = 2A_{2,0} - A_{1,0}$	$A_{1,2} = \frac{4A_{2,1} - A_{1,1}}{3}$
$A_{2,0} = P(2)$	$A_{2,1} = 2A_{3,0} - A_{2,0}$	
$A_{3,0} = P(4)$		

Table 2.4.1: Approximation of an American option price with 3 Bermudan options

Chapter 3

Numerical Solutions of P(I)DEs

3.1 Numerical Solution of Black-Scholes' PDE

Next to the quadrate methods described in the previous section, standard numerical techniques can also be used to approximate the option price V. The techniques used in this thesis for equation (1.1.3) are known as $\mathcal{O}\left(h^2\right)$ central differences for the derivatives in the spatial directions and the Crank-Nicolson method for the time derivative. Details on these methods can be found in any textbook on finite differences, e.g. Burden and Faires (2001). Define by $h = \frac{U-L}{N-1}$ the step size in spatial direction, with N the number of points, and $k = \frac{T-t_0}{M}$, with M the number of time steps, then

$$\begin{aligned} \frac{\partial^2 V}{\partial x^2} &= \frac{V_{i-1} - 2V_i + V_{i+1}}{h^2} + \mathcal{O}\left(h^2\right), \\ \frac{\partial V}{\partial x} &= \frac{V_{i+1} - V_{i-1}}{2h} + \mathcal{O}\left(h^2\right), \end{aligned}$$

where $V_i = V\left(x_i, \tau\right)$, with $x_i = L + (i-1)\,h$ for $i = 1, ..., N$. Note that (1.1.3) can be simplified as $\frac{\partial V}{\partial \tau} = g\left(x, \tau\right)$, where $g\left(x, \tau\right)$ is the right-hand side of the equation. The Crank-Nicolson method approximates (1.1.3) as

$$\frac{V_i^{m+1} - V_i^m}{k} = \frac{1}{2}\left(g\left(x, \tau_{m+1}\right) + g\left(x, \tau_m\right)\right) + \mathcal{O}\left(h^2\right),$$

where $V_i^m = V\left(x_i, \tau_m\right)$, with $\tau_m = (m-1)\,k$ for $m = 1, ..., M$. This results in the following system of equations:

$$\mathbf{A}\mathbf{V}^{m+1} = \mathbf{b}^m, \tag{3.1.1}$$

with

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ \alpha & \beta & \gamma & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \alpha & \beta & \gamma \\ 0 & \cdots & 0 & 0 & 1 \end{bmatrix}, \mathbf{V}^m = \begin{bmatrix} V_1^m \\ \vdots \\ \vdots \\ \vdots \\ V_N^m \end{bmatrix}$$

and

$$\mathbf{b}^m = \begin{bmatrix} l \\ V_2^m + \frac{1}{2}k\left(\sigma^2 \frac{V_1^m - 2V_2^m + V_3^m}{2h^2} + \left(r - \frac{1}{2}\sigma^2\right)\frac{V_3^m - V_1^m}{2h} - rV_2^m\right) \\ \vdots \\ V_{N-1}^m + \frac{1}{2}k\left(\sigma^2 \frac{V_{N-2}^m - 2V_{N-1}^m + V_N^m}{2h^2} + \left(r - \frac{1}{2}\sigma^2\right)\frac{V_N^m - V_{N-2}^m}{2h} - rV_{N-1}^m\right) \\ u \end{bmatrix}.$$

Here $\alpha = \frac{\left(r-\frac{1}{2}\sigma^2\right)kh - \sigma^2 k}{4h^2}$, $\beta = 1 + \frac{\sigma^2 k + rkh^2}{2h^2}$ and $\gamma = \frac{-\left(r-\frac{1}{2}\sigma^2\right)kh - \sigma^2 k}{4h^2}$. l and u denote the lower and upper boundary condition, respectively. For example, $l = Ee^{-r\tau_m}$ and $u = 0$ for a European put option. The initial condition $\mathbf{V}^1$ represents the payoff.

Solving system (3.1.1) M times gives the value of a European option. To price an American option one can use the projected SOR iterative algorithm as described in Wilmott (1998). This algorithm evaluates for each element of $\mathbf{V}^{m+1}$ whether the solution is below the payoff, e.g. for an American put option $V_i^{m+1} > \max\left(K - e^{x_i}, 0\right)$. If this is not the case V_i^{m+1} is projected, i.e. is set equal to the payoff.

3.2 Numerical Solution of Heston's PDE

The PDE for the Heston's model is somewhat more complicated than the PDE in the Black-Scholes world. This is because it contains the variance, v. The Heston PDE is discretized by the same discretization rules as the Black-Scholes PDE, i.e. $\mathcal{O}\left(h^2\right)$ central differences in spatial and volatility directions and Crank-Nicolson in time direction. So,

$$\frac{\partial^2 V}{\partial x^2} = \frac{V_{i-1,j} - 2V_{i,j} + V_{i+1,j}}{\Delta x^2} + \mathcal{O}\left(\Delta x^2\right), \quad \frac{\partial^2 V}{\partial v^2} = \frac{V_{i,j-1} - 2V_{i,j} + V_{i,j+1}}{\Delta v^2} + \mathcal{O}\left(\Delta v^2\right),$$
$$\frac{\partial V}{\partial x} = \frac{V_{i+1,j} - V_{i-1,j}}{2\Delta x} + \mathcal{O}\left(\Delta x^2\right), \qquad \frac{\partial V}{\partial v} = \frac{V_{i,j+1} - V_{i,j-1}}{2\Delta v} + \mathcal{O}\left(\Delta v^2\right),$$

$$\frac{\partial^2 V}{\partial x \partial v} = \frac{V_{i+1,j+1} - V_{i+1,j-1} - V_{i-1,j+1} + V_{i-1,j-1}}{4\Delta x \Delta v} + \mathcal{O}\left(\Delta x^2\right) + \mathcal{O}\left(\Delta v^2\right),$$

where $\Delta x = \frac{U-L}{N-1}$, with N the number of points in x-direction, $\Delta v = \frac{v^*}{L-1}$, with L the number of points in v-direction, and $V_{i,j} = V\left(x_i, v_j, \tau\right)$, with $x_i = L + (i-1)\Delta x$ for $i = 1, ..., N$ and $v_j = (j-1)\Delta v$ for $j = 1, ..., L$. Using Crank-Nicolson, as described in section 3.1, this results in the following system of equations:

$$\mathbf{A}\mathbf{V}^{m+1} = \mathbf{b}^m, \tag{3.2.1}$$

where $\mathbf{A}$ a block tridiagonal matrix of the form

$$\mathbf{A} = \begin{bmatrix} I & 0 & 0 & \cdots & 0 \\ L_2 & D_2 & U_2 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & L_{L-1} & D_{L-1} & U_{L-1} \\ 0 & \cdots & 0 & 0 & I \end{bmatrix},$$

where I is the identity matrix and the other blocks read

$$D_j = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ \alpha_j & \beta_j & \gamma_j & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \alpha_j & \beta_j & \gamma_j \\ 0 & \cdots & 0 & 0 & 1 \end{bmatrix}, \text{ for } j = 2, ..., L-1,$$

with $\alpha_j = \frac{r\Delta x - \left(1+\frac{1}{2}\Delta x\right)v_j}{4(\Delta x)^2}\Delta t$, $\beta_j = 1 + \left(\frac{v_j}{2(\Delta x)^2} + \frac{v_j\eta^2}{2(\Delta v)^2} + \frac{1}{2}r\right)\Delta t$, $\gamma_j = \frac{\left(\frac{1}{2}\Delta x - 1\right)v_j - r\Delta x}{4(\Delta x)^2}\Delta t$, and

$$U_j = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ \delta_j & \varepsilon_j & -\delta_j & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \delta_j & \varepsilon_j & -\delta_j \\ 0 & \cdots & 0 & 0 & 0 \end{bmatrix}, L_j = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ -\delta_j & \zeta_j & \delta_j & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -\delta_j & \zeta_j & \delta_j \\ 0 & \cdots & 0 & 0 & 0 \end{bmatrix}, \text{ for } j = 2, ..., L-1,$$

where $\delta_j = \frac{\rho\eta v_j}{8\Delta x\Delta v}\Delta t$, $\varepsilon_j = \frac{\left(\lambda\Delta v - \eta^2\right)v_j - \lambda\overline{v}\Delta v}{4(\Delta v)^2}\Delta t$ and $\zeta_j = \frac{\lambda\overline{v}\Delta v - \left(\eta^2 + \lambda\Delta v\right)v_j}{4(\Delta v)^2}\Delta t$. Each block is of size $N \times N$. Notice that each first and last row of a block-row contains only a one on the diagonal. This is because of the non-eliminated boundary conditions.

The vector $\mathbf{V}^m$ is defined by $\mathbf{V}^m = \begin{bmatrix} V^m_{1,1} & \cdots & V^m_{N,1} & \cdots & \cdots & V^m_{1,M} & \cdots & V^m_{N,M} \end{bmatrix}^T$ and $\mathbf{b}^m$ equals $\mathbf{b}^m = \begin{bmatrix} B_1 & B_2 & \cdots & B_{L-1} & B_L \end{bmatrix}^T$, where B_1, B_L form the lower and upper boundary condition of the volatility, respectively. Further, for $j = 2, ..., L-1$ we have $B_j = \begin{bmatrix} l & \theta^m_{2,j} & \theta^m_{N-1,j} & u \end{bmatrix}^T$, with

$$\begin{aligned} \theta^m_{i.j} &= V^m_{i,j} + \frac{\Delta t}{2}\left(\frac{1}{2}v_j\frac{V^m_{i-1,j} - 2V^m_{i,j} + V^m_{i+1,j}}{(\Delta x)^2} + \rho\eta v_j\frac{V^m_{i+1,j+1} - V^m_{i+1,j-1} - V^m_{i-1,j+1} + V^m_{i-1,j-1}}{4\Delta x\Delta v}\right. \\ &\quad \left. + \frac{1}{2}v_j\eta^2\frac{V^m_{i,j-1} - 2V^m_{i,j} + V^m_{i,j+1}}{(\Delta v)^2} + \left(r - \frac{1}{2}v_j\right)\frac{V^m_{i+1,j} - V^m_{i-1,j}}{2\Delta x} - \lambda\left(v_j - \overline{v}\right)\frac{V^m_{i,j+1} - V^m_{i,j-1}}{2\Delta v} - rV^m_{i,j}\right). \end{aligned}$$

The elements l and u are the boundary conditions for $x = L$ and $x = U$, respectively. For example, for a European call option we have: $l = 0$, $u = e^{x^*} - K$, $B_1 = \begin{bmatrix} (e^{x_1} - K)^+ & \cdots & (e^{x_N} - K)^+ \end{bmatrix}^T$ and $B_M = \begin{bmatrix} e^{x_1} & \cdots & e^{x_N} \end{bmatrix}^T$.

To visualize the present material we consider a simple example.

Example 1 *Assume $M = 4$, $N = 4$ and V is the price of a European call option. The matrix* $\mathbf{A}$*, the unknown vector* $\mathbf{V}^{m+1}$ *and the right hand side* $\mathbf{b}^m$ *will have the form*

$$\mathbf{A} = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-\delta_2 & \zeta_2 & \delta_2 & 0 & \alpha_2 & \beta_2 & \gamma_2 & 0 & \delta_j & \varepsilon_2 & -\delta_2 & 0 & 0 & 0 & 0 & 0 \\
0 & -\delta_2 & \zeta_2 & \delta_2 & 0 & \alpha_2 & \beta_2 & \gamma_2 & 0 & \delta_2 & \varepsilon_2 & -\delta_2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -\delta_3 & \zeta_3 & \delta_3 & 0 & \alpha_3 & \beta_3 & \gamma_3 & 0 & \delta_3 & \varepsilon_3 & -\delta_3 & 0 \\
0 & 0 & 0 & 0 & 0 & -\delta_3 & \zeta_3 & \delta_3 & 0 & \alpha_3 & \beta_3 & \gamma_3 & 0 & \delta_3 & \varepsilon_3 & -\delta_3 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix},$$

$$\mathbf{V}^{m+1} = \begin{bmatrix}
V_{1,1}^{m+1} \\ V_{2,1}^{m+1} \\ V_{3,1}^{m+1} \\ V_{4,1}^{m+1} \\ V_{1,2}^{m+1} \\ V_{2,2}^{m+1} \\ V_{3,2}^{m+1} \\ V_{4,2}^{m+1} \\ V_{1,3}^{m+1} \\ V_{2,3}^{m+1} \\ V_{3,3}^{m+1} \\ V_{4,3}^{m+1} \\ V_{1,4}^{m+1} \\ V_{2,4}^{m+1} \\ V_{3,4}^{m+1} \\ V_{4,4}^{m+1}
\end{bmatrix} \quad \textit{and } \mathbf{b}^m = \begin{bmatrix}
\left(e^{-x^*} - K\right)^+ \\ \left(e^{x_2} - K\right)^+ \\ \left(e^{x_3} - K\right)^+ \\ \left(e^{x^*} - K\right)^+ \\ 0 \\ \theta_{2,2} \\ \theta_{3,2} \\ e^{x^*} - K \\ 0 \\ \theta_{2,3} \\ \theta_{3,3} \\ e^{x^*} - K \\ e^{-x^*} \\ e^{x_2} \\ e^{x_3} \\ e^{x^*}
\end{bmatrix}.$$

3.3 Numerical Solution of the Variance Gamma PIDE

The method described in this section originates from Almendral and Oosterlee (2006). The challenge of solving this PIDE numerically lies in the integral, which is undefined at $x = 0$. Recall that the PIDE reads

$$\frac{\partial V}{\partial \tau} - \mathcal{L}V = 0,$$

where

$$\mathcal{L}\varphi = r\varphi_x - r\varphi + \int_{-\infty}^{\infty} \left[\varphi\left(x + y\right) - \varphi\left(x\right) - \left(e^y - 1\right)\varphi_x\right] k\left(y\right) dy.$$

The idea of the method is to consider one part of the integral term implicitly and the remaining parts explicitly.

Consider a computational domain of the form $[0,T]\times[L,U]$. Let the time interval be divided into M equal parts: $0=t_0<t_1<\cdots<t_M=T$, with $t_m=mk$, $m=0,1,...,M$ and $k=\frac{T}{M}$. The spatial interval $[L,U]$ contains the point $\ln K$, and $L=x_1<x_2<\cdots<x_N=U$, with $x_n=L+(n-1)\,h$, $n=1,...,N$, and h is such that $h=\frac{U-L}{N-1}$.

We split the operator $\mathcal{L}$ into a sum of two operators $\mathcal{A}$ and $\mathcal{B}$:

$$\mathcal{A}\varphi:=(r+\omega(h))\,\varphi_x-r\varphi+\int\limits_{|y|\le h}\left[\varphi(x+y)-\varphi(x)-(e^y-1)\,\varphi_x\right]k(y)\,dy$$

with

$$\omega(h)=\int\limits_{|y|\ge h}(1-e^y)\,k(y)\,dy$$

and

$$\mathcal{B}\varphi:=\int\limits_{|y|\ge h}\left[\varphi(x+y)-\varphi(x)\right]k(y)\,dy.$$

In order to have a method that is second order accurate in time, the well-known BDF2 scheme is proposed. The second order timestepping method reads

$$\frac{\frac{3}{2}V^{m+1}-2V^m+\frac{1}{2}V^{m-1}}{k}-\mathcal{A}V^{m+1}-\mathcal{B}V^m=0. \tag{3.3.1}$$

3.3.1 Spatial Discretization of $\mathcal{A}$

The idea is here to approximate all the integrals to $\mathcal{O}\left(h^2\right)$, in the presence of the singular density:

$$k(y)=\mathcal{O}\left(\tfrac{1}{y}\right),\quad y\to 0. \tag{3.3.2}$$

Let us first look at the positive part of the integral:

$$\begin{aligned}
&\int\limits_0^h\left[\varphi(x+y)-\varphi(x)-(e^y-1)\,\varphi_x\right]k(y)\,dy\\
=&\int\limits_0^h\left[\varphi(x+y)-\varphi(x)-y\varphi_x(x)-(e^y-1-y)\,\varphi_x\right]k(y)\,dy\\
=&\int\limits_0^h\left(\varphi(x+y)-\varphi(x)-y\varphi_x(x)\right)k(y)\,dy-\varphi_x\int\limits_0^h(e^y-1-y)\,k(y)\,dy.
\end{aligned} \tag{3.3.3}$$

Recall from Taylor that

$$e^y-1-y=\tfrac{1}{2}y^2+\mathcal{O}\left(y^3\right)$$

and

$$\varphi(x+y)-\varphi(x)-y\varphi_x(x)=\tfrac{1}{2}y^2\varphi_{xx}(x)+\mathcal{O}\left(y^3\right).$$

Therefore, the trapezoidal rule and equation (3.3.2) applied to (3.3.3) gives

$$\begin{aligned}
&\int\limits_0^h\left[\varphi(x+y)-\varphi(x)-(e^y-1)\,\varphi_x\right]k(y)\,dy\\
=&\frac{\varphi_{xx}}{2}\int\limits_0^h\left(y^2+\mathcal{O}\left(y^3\right)\right)k(y)\,dy-\frac{\varphi_x}{2}\int\limits_0^h\left(y^2+\mathcal{O}\left(y^3\right)\right)k(y)\,dy\\
=&\frac{\varphi_{xx}}{2}\int\limits_0^h y+\mathcal{O}\left(y^2\right)dy-\frac{\varphi_x}{2}\int\limits_0^h y+\mathcal{O}\left(y^2\right)dy\\
=&\frac{\varphi_{xx}}{4}h^2-\frac{\varphi_x}{4}h^2+\mathcal{O}\left(h^3\right)
\end{aligned}$$

Similar to the positive part of the integral, the negative part can be written as

$$\begin{aligned} & \int_{-h}^{0} \left[\varphi\left(x+y\right)-\varphi\left(x\right)-\left(e^{y}-1\right)\varphi_{x}\right] k\left(y\right) dy \\ = \quad & -\frac{\varphi_{xx}}{4}h^{2}+\frac{\varphi_{x}}{4}h^{2}+\mathcal{O}\left(h^{3}\right) \end{aligned}$$

So, the sum of the positive and negative parts is of $\mathcal{O}\left(h^{2}\right)$. Therefore, dropping the term

$$\int_{|y|\leq h} \left[\varphi\left(x+y\right)-\varphi\left(x\right)-\left(e^{y}-1\right)\varphi_{x}\right] k\left(y\right) dy$$

does not affect the accuracy of the scheme. The remaining integral of $\mathcal{A}$, $\omega\left(h\right)$, is discretized by the trapezoidal rule and the partial derivatives by second order accurate central differences.

3.3.2 Spatial Discretization of $\mathcal{B}$

Away from the origin, the integral term in $\mathcal{B}$ may be split into a sum of two terms:

$$\int_{|y|\geq h} \left[\varphi\left(x+y\right)-\varphi\left(x\right)\right] k\left(y\right) dy = J\left(x\right)-\varphi\left(x\right)\lambda\left(h\right), \tag{3.3.4}$$

with

$$J\left(x\right) = \int_{|y|\geq h} \varphi\left(x+y\right) k\left(y\right) dy, \tag{3.3.5}$$

$$\lambda\left(h\right) = \int_{|y|\geq h} k\left(y\right) dy. \tag{3.3.6}$$

Both integrals are discretized by the trapezoidal rule to obtain second order accuracy. Firstly consider integral $\lambda\left(h\right)$:

$$\lambda\left(h\right) \approx h\sum_{l=1}^{N} k\left(y_{l}\right)\rho_{l}, \tag{3.3.7}$$

where $y_{l}=x_{l}$, $\rho_{l}=\frac{1}{2}$ at the boundaries and $\rho_{l}=1$ for the remaining l, and $k\left(0\right)$ is redefined as 0. Notice that the integrals $\lambda\left(h\right)$ and $\omega\left(h\right)$ can be approximated by any integration rule, e.g. the Gaussian quadrature routine, because the integrand is known analytically.

The other integral, $J\left(x\right)$, is more difficult to approximate. This is because the integrand is only known at fixed grid-points as specified in section 3.3. For a particular grid-point x_{n} we have

$$J\left(x_{n}\right) = \int_{|y|\geq h} \varphi\left(x_{n}+y\right) k\left(y\right) dy.$$

Applying the trapezoidal rule results in

$$J\left(x_{n}\right) \approx h\sum_{l=1}^{N} \varphi\left(x_{n}+y_{l}\right) k\left(y_{l}\right)\rho_{l}. \tag{3.3.8}$$

In case $x_{n}+y_{l}\leq L$ or $x_{n}+y_{l}\geq U$ something special needs to be done. A European call option, for example, approaches $e^{x}-Ke^{-r(T-t)}$ for x large, and 0 for x large negative, φ is then approximated by $\varphi\left(x\right)=e^{x}-Ke^{-r(T-t)}$ if $x\geq U$ and by $\varphi\left(x\right)=0$ if $x\leq L$. Notice that $\mathbf{J}=\left[\begin{array}{ccc} J\left(x_{1}\right) & \cdots & J\left(x_{N}\right)\end{array}\right]^{T}$ can be written as a matrix product:

$$\mathbf{J} = h\mathbf{\Phi}\mathbf{k}, \tag{3.3.9}$$

where

$$\mathbf{\Phi} = \begin{bmatrix} \varphi(x_1 + y_1) & \cdots & \varphi(x_1 + y_N) \\ \vdots & & \vdots \\ \varphi(x_N + y_1) & \cdots & \varphi(x_N + y_N) \end{bmatrix} \text{ and } \mathbf{k} = \begin{bmatrix} k(y_1)\rho_1 \\ \vdots \\ k(y_N)\rho_N \end{bmatrix}.$$

In general the calculation of such a product costs $\mathcal{O}(N^2)$ operations. For this particular example, the computational cost can be reduced to $\mathcal{O}(N \log N)$ by using the FFT algorithm. This algorithm is applicable if the matrix is a Toeplitz matrix. A Toeplitz matrix is a matrix which is constant along its diagonals. The next section will present the application of FFT to Toeplitz matrices. We transform the matrix Φ into a Toeplitz matrix, which can be done by a simple transformation:

$$\mathbf{J} = h\widetilde{\mathbf{\Phi}}\widetilde{\mathbf{k}}, \tag{3.3.10}$$

where

$$\widetilde{\mathbf{\Phi}} = \begin{bmatrix} \varphi(x_1 + y_N) & \cdots & \varphi(x_1 + y_1) \\ \vdots & & \vdots \\ \varphi(x_N + y_N) & \cdots & \varphi(x_N + y_1) \end{bmatrix} \text{ and } \widetilde{\mathbf{k}} = \begin{bmatrix} k(y_N)\rho_N \\ \vdots \\ k(y_1)\rho_1 \end{bmatrix}.$$

Lemma 1 *The matrix $\widetilde{\mathbf{\Phi}}$ in (3.3.10) is a Toeplitz matrix.*

Proof. A Toeplitz matrix is constant along its diagonals. In other words, for an arbitrary element $\widetilde{\mathbf{\Phi}}_{i,j}$ of $\widetilde{\mathbf{\Phi}}$ with $i, j = 1, ..., N-1$ it must hold that $\widetilde{\mathbf{\Phi}}_{i,j} = \widetilde{\mathbf{\Phi}}_{i+1,j+1}$:

$$\begin{aligned} \widetilde{\mathbf{\Phi}}_{i,j} &= \varphi\left(x_i + y_{N-(j-1)}\right) \\ &= \varphi(L + (i-1)h + L + (N - (j-1) - 1)h) \\ &= \varphi(2L + (i-1)h + (N-j)h) \\ &= \varphi(2L + (N + i - j - 1)h) \\ &= \varphi(2L + (N + (i+1) - (j+1) - 1)h) \\ &= \varphi(L + ((i+1) - 1)h + L + (N - j - 1)h) \\ &= \varphi(x_{i+1} + y_{N-j}) = \widetilde{\mathbf{\Phi}}_{i+1,j+1}. \end{aligned}$$

■

3.3.3 Application of FFT to Toeplitz Matrices

This section describes the way the FFT algorithm can be applied to Toeplitz matrices, see e.g. Almendral and Oosterlee (2005). The FFT algorithm cannot be applied to Toeplitz matrices directly. Firstly, the Toeplitz matrix needs to be embedded into a circulant matrix, and the product of a circulant matrix and a vector may be efficiently computed by applying the FFT algorithm. Each row in a circulant matrix is by definition a circular shift of the previous row. For example, the following matrix is a circulant 3×3 matrix:

$$\begin{bmatrix} a_1 & a_2 & a_3 \\ a_3 & a_1 & a_2 \\ a_2 & a_3 & a_1 \end{bmatrix}. \tag{3.3.11}$$

Now, consider the following Toeplitz matrix:

$$\mathbf{T} = \begin{bmatrix} a_0 & a_1 & \cdots & a_n \\ a_{2n} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & a_1 \\ a_{n+1} & \cdots & a_{2n} & a_0 \end{bmatrix}. \tag{3.3.12}$$

This matrix can be embedded in a circulant matrix of size $2n+1$:

$$\mathbf{C} = \begin{bmatrix} a_0 & a_1 & \cdots & a_n & a_{n+1} & \cdots & a_{2n} \\ a_{2n} & a_0 & \ddots & & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & & \ddots & a_{n+1} \\ a_{n+1} & & \ddots & \ddots & \ddots & & a_n \\ a_n & \ddots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & & \ddots & a_0 & a_1 \\ a_1 & \cdots & a_n & a_{n+1} & \cdots & a_{2n} & a_0 \end{bmatrix}. \tag{3.3.13}$$

If we define the vectors $\mathbf{d} = \begin{bmatrix} d_0 & \cdots & d_n \end{bmatrix}^T$ and $\mathbf{f} = \begin{bmatrix} d_0 & \cdots & d_n & 0 & \cdots & 0 \end{bmatrix}^T$, then the product $\mathbf{Td}$ is the vector consisting of the first $(n+1)$ elements of the product $\mathbf{Cf}$. Thus, as soon $\mathbf{Cf}$ is calculated the last n elements need to be deleted. The i-th element of the resulting vector can be written as

$$(\mathbf{Cf})_i = \sum_{j=0}^{2n} a_{j-i} d_j, \tag{3.3.14}$$

which is equal to the definition of a convolution, $\mathbf{Cf} = \mathbf{a} * \mathbf{f}$. An important application of the Fourier transform is in convolutions. Convolutions can be evaluated by component-wise multiplications in Fourier space. So,

$$\widehat{\mathbf{Cf}} = \widehat{\mathbf{a}} \cdot \widehat{\mathbf{f}},$$

where $\widehat{\mathbf{x}}$ is the Fourier transform of an arbitrary vector $\mathbf{x}$ and the multiplication is done component-wise. Thus, the multiplication of the Toeplitz matrix $\mathbf{T}$ and the vector $\mathbf{d}$ is given by the first $(n+1)$ elements of the inverse Fourier transform of $\widehat{\mathbf{Cf}}$.

Chapter 4

Monte Carlo Simulation

Monte Carlo methods are applicable to a wide range of problems. In this chapter we focus on derivative pricing. Boyle (1977) was the first to use these methods for the valuation of options. The method can roughly be described as follows. Firstly, many paths of the underlying are simulated. Then the payoff can be determined for each simulation and the price of the option is the discounted average of the simulated payoffs. The Monte Carlo method is applicable to many payoffs. It can especially handle path dependent options very well, which is the main advantage of the method, together with the ability to solve high-dimensional problems. The method is less suited for callable derivatives. Though, with the celebrated Longstaff-Schwartz algorithm, see (Longstaff and Schwartz, 2001), callable features can be handled if necessary. This section discusses the Monte Carlo technique and presents methods to simulate paths. Also a brief description of a variance reduction technique called antithetic variates is found in this chapter.

Consider a payoff function g, which is dependent on the path of the underlying asset $(x_t)_{t_0 \leq t \leq T}$, then today's value of the derivative with payoff g at maturity reads

$$V = e^{-r(T-t_0)} \mathbb{E}_{t_0} \left[g\left((x_t)_{t_0 \leq t \leq T} \right) \right]. \tag{4.0.1}$$

The Monte Carlo method to solve this can be summarized as follows:

1. Establish a procedure to simulate a path given the dynamics of the underlying model.
2. Simulate a path $\left(y_t^i\right)_{t_0 \leq t \leq T}$.
3. Compute the payoff $g\left(\left(y_t^i\right)_{t_0 \leq t \leq T}\right)$ and store the result.
4. Repeat step 2 and 3 for $i = 1, ..., N$ and compute the average:

$$\overline{\mu}\left(\left(y_t^1\right)_{t_0 \leq t \leq T}, \cdots, \left(y_t^N\right)_{t_0 \leq t \leq T}\right) = \frac{1}{N} \sum_{i=1}^{N} g\left(\left(y_t^i\right)_{t_0 \leq t \leq T}\right). \tag{4.0.2}$$

   Here $\overline{\mu}$ is known as the Monte Carlo estimate.

5. Calculate the option value as

$$V = e^{-r(T-t_0)} \overline{\mu}. \tag{4.0.3}$$

The way these paths can be simulated is presented in the following sections.

4.1 Antithetic Variates

Antithetic Variates (AV) represents a very simple, and widely used, variance reduction technique for a Monte Carlo simulation. The variance reduction is obtained by drawing antithetic pairs $\{z,-z\}$ from the Gaussian probability distribution. Notice that if Z is normally distributed, so is $-Z$. As a result, $\frac{N}{2}$ antithetic pairs are more regularly distributed than N independent Gaussian variables, e.g. the mean is always zero, which almost surely does not hold for the latter. This may lead to a reduction in variance, see e.g. Buitelaar (2006). We replace our estimate $\overline{\mu}$ in equation (4.0.2) and (4.0.3) by

$$\overline{\mu}_{AV}\left(\left(y_t^1\right)_{t_0\leq t\leq T},\cdots,\left(y_t^{\frac{N}{2}}\right)_{t_0\leq t\leq T},\left(y_t^{-1}\right)_{t_0\leq t\leq T},\cdots,\left(y_t^{-\frac{N}{2}}\right)_{t_0\leq t\leq T}\right)=\frac{\overline{\mu}^{+}+\overline{\mu}^{-}}{2}, \quad (4.1.1)$$

where $\overline{\mu}^{+}=\overline{\mu}\left(\left(y_t^1\right)_{t_0\leq t\leq T},\cdots,\left(y_t^{\frac{N}{2}}\right)_{t_0\leq t\leq T}\right)$ and $\overline{\mu}^{-}=\overline{\mu}\left(\left(y_t^{-1}\right)_{t_0\leq t\leq T},\cdots,\left(y_t^{-\frac{N}{2}}\right)_{t_0\leq t\leq T}\right)$. Here $\left(y_t^i\right)_{t_0\leq t\leq T}$, for $i=1,...,\frac{N}{2}$, are the paths based on Z and $\left(y_t^{-i}\right)_{t_0\leq t\leq T}$, for $i=1,...,\frac{N}{2}$, are the paths based on $-Z$.

4.2 The Brownian Motion with Drift

The stochastic differential equation of a Brownian motion with drift is defined as

$$\begin{aligned} dx(t) &= \mu(x(t),t)\,dt+\sigma(x(t),t)\,dW(t), \quad (4.2.1)\\ x(t_0) &= x_0. \end{aligned}$$

Notice that the Black-Scholes dynamics and the local volatility dynamics are special cases of (4.2.1). The solution of (4.2.1) reads

$$x(T)=x_0+\int_{t_0}^{T}\mu(x(t),t)\,dt+\int_{t_0}^{T}\sigma(x(t),t)\,dW(t) \quad (4.2.2)$$

In some models analytic solutions are available. In its general form, (4.2.2), an explicit solution cannot be derived. This is because the functions $\mu(x(t),t)$ and $\sigma(x(t),t)$ are not known explicitly. So, integrals need to be solved numerically. Numerical integration methods for stochastic integrals are discussed in the following sections.

Consider an equidistant grid in time direction with $t_m=t_0+m\Delta t$ for $m=0,...,M$, and $\Delta t=t_m-t_{m-1}=\frac{T-t_0}{M}$. Then for arbitrary $m\in\{1,...,M\}$ one has

$$x_m=x_{m-1}+\int_{t_{m-1}}^{t_m}\mu(x(t),t)\,dt+\int_{t_{m-1}}^{t_m}\sigma(x(t),t)\,dW(t)$$

where $x_m=x(t_m)$. This follows from Itô's lemma and the Markov property of (4.2.1). So, we can define the following algorithm.

Algorithm 1

1. *x_0 is known*
2. $x_m=x_{m-1}+\int_{t_{m-1}}^{t_m}\mu(x(t),t)\,dt+\int_{t_{m-1}}^{t_m}\sigma(x(t),t)\,dW(t)$, *for* $m=1,...,M$

Numerical integration schemes are used to approximate the integrals in algorithm 1. The order of convergence of such a scheme is defined as follows:

Definition 2 *(Strong Convergence) Denote $y^{\Delta t}(T)$ as the approximation for $x(T)$, where Δt is the step size. Then $y^{\Delta t}(T)$ converges strongly to $x(T)$ with order $\gamma > 0$ if*

$$\mathbb{E}\left|y^{\Delta t}(T) - x(T)\right| = \mathcal{O}\left((\Delta t)^{\gamma}\right).$$

Definition 3 *(Weak Convergence) For some function g, $y^{\Delta t}(T)$ converges weakly to $x(T)$ with respect to g with order $\beta > 0$ if*

$$\left|\mathbb{E}\left[g\left(y^{\Delta t}(T)\right)\right] - \mathbb{E}\left[g\left(x(T)\right)\right]\right| = \mathcal{O}\left((\Delta t)^{\beta}\right).$$

In words we could say that a numerical integration method converges strongly if the whole path converges, whereas weak convergence only implies a convergent approximation of the probability distribution of $x(T)$.

4.2.1 Stochastic Euler and Milstein Scheme

Stochastic Euler, or Euler-Maruyama, is a numerical integration method for stochastic integrals. This integration method is defined as follows:

$$\begin{aligned} y_m^{\Delta t} &= y_{m-1}^{\Delta t} + \int_{t_{m-1}}^{t_m} \mu\left(y_{m-1}^{\Delta t}, t_{m-1}\right) dt + \int_{t_{m-1}}^{t_m} \sigma\left(y_{m-1}^{\Delta t}, t_{m-1}\right) dW(t), \\ &= y_{m-1}^{\Delta t} + \mu\left(y_{m-1}^{\Delta t}, t_{m-1}\right) \int_{t_{m-1}}^{t_m} dt + \sigma\left(y_{m-1}^{\Delta t}, t_{m-1}\right) \int_{t_{m-1}}^{t_m} dW(t), \\ &= y_{m-1}^{\Delta t} + \mu\left(y_{m-1}^{\Delta t}, t_{m-1}\right) \Delta t + \sigma\left(y_{m-1}^{\Delta t}, t_{m-1}\right) W(\Delta t). \end{aligned} \tag{4.2.3}$$

where $y_m^{\Delta t}$ denotes the approximation of x_m and $y_0^{\Delta t} = x_0$. $W(\Delta t)$ is normally distributed with mean zero and variance Δt, $W(\Delta t) \sim N(0, \Delta t)$. An outcome $w_{\Delta t}$ from this distribution is obtained by computing $w_{\Delta t} = z\sqrt{\Delta t}$, where z is drawn from a standard normal distribution. The Euler-Maruyama scheme converges strongly with order $1/2$ and weakly with order 1 under some naturally fulfilled conditions for the Black-Scholes model and the local volatility model. For a complete proof we refer to Kloeden and Platen (1992), Chapter 10 and Chapter 14.

In case of deterministic differential equations one can use Taylor expansions to obtain a higher order of convergence. For stochastic differential equations a similar approach exists, which uses the stochastic Taylor expansion, or Itô-Taylor expansion. The stochastic Euler approximation only uses the first 2 terms of this expansion. The Milstein scheme is obtained by adding a third term:

$$\begin{aligned} y_m^{\Delta t} &= y_{m-1}^{\Delta t} + \mu\left(y_{m-1}^{\Delta t}, t_{m-1}\right) \Delta t + \sigma\left(y_{m-1}^{\Delta t}, t_{m-1}\right) W(\Delta t) \\ &\quad + \tfrac{1}{2}\sigma\left(y_{m-1}^{\Delta t}, t_{m-1}\right) \frac{\partial \sigma}{\partial x}\left(y_{m-1}^{\Delta t}, t_{m-1}\right) \left(W(\Delta t)^2 - \Delta t\right), \\ y_0^{\Delta t} &= x_0. \end{aligned} \tag{4.2.4}$$

For the Black-Scholes and the local volatility model this scheme converges both strongly as weakly with order 1. For a proof we refer to Chapter 10 and Chapter 14 of Kloeden and Platen (1992). The application of the Millstein scheme to multi-dimensional models is not straightforward at all. More on this can be found in Kloeden and Platen (1992), but we suggest not to use it at all for multi-dimensional models.

4.3 A Simulation Scheme for Heston's Model

In this section we provide a method to simulate logarithmic stock prices under the Heston model. The method originates from Lord, Koekkoek and van Dijk (2006), in which several methodologies are compared. Their best performing method is presented here. For reasons of clarity, we repeat equation (1.2.1) in a way, that specifies the dynamics of logarithmic stock prices and the variance process in the Heston model under the risk neutral probability measure:

$$dx(t) = \left(r - \frac{1}{2}v(t)\right)dt + \rho\sqrt{v(t)}dW_1(t) + \sqrt{1-\rho^2}\sqrt{v(t)}dW_2(t), \tag{4.3.1}$$

$$dv(t) = -\lambda(v(t) - \overline{v})\,dt + \eta\sqrt{v(t)}dW_1(t). \tag{4.3.2}$$

With the Euler discretization to simulate the variance one encounters into the problem that, whereas the mean-reverting variance process is guaranteed to be non-negative, the Euler discretization is not. This can easily be verified as follows. As the mean-reverting variance process is continuous, in order for it to become negative it must reach zero as well. Notice that equation (4.3.2) is completely deterministic for $v(t) = 0$. So, as soon the variance reaches zero it will directly be positive afterwards. In contrast the Euler discretization is not continuous. In other words, the discretized form can become immediately negative. In order to prevent the volatility to cross over the imaginary axis, we have to decide what to do with negative variance. The so-called full truncation scheme fixes this problem here.

The solution to the two dimensional SDE, (4.3.1) and (4.3.2), reads

$$x(T) = x(t_0) + \int_{t_0}^{T}\left(r - \frac{1}{2}v(t)\right)dt + \rho\int_{t_0}^{T}\sqrt{v(t)}dW_1(t) + \sqrt{1-\rho^2}\int_{t_0}^{T}\sqrt{v(t)}dW_2(t), \tag{4.3.3}$$

$$v(T) = v(t_0) - \lambda\int_{t_0}^{T}(v(t) - \overline{v})\,dt + \eta\int_{t_0}^{T}\sqrt{v(t)}dW_1(t). \tag{4.3.4}$$

Now, if we consider an equidistant grid in time direction with $t_m = t_0 + m\Delta t$ for $m = 0, ..., M$, and $\Delta t = t_m - t_{m-1} = \frac{T-t_0}{M}$, then we can employ the following algorithm to simulate a path.

Algorithm 1

1. *x_0 and v_0 are known*
2. $x_m = x_{m-1} + \int_{t_{m-1}}^{t_m}\left(r - \frac{1}{2}v(t)\right)dt + \rho\int_{t_{m-1}}^{t_m}\sqrt{v(t)}dW_1(t) + \sqrt{1-\rho^2}\int_{t_{m-1}}^{t_m}\sqrt{v(t)}dW_2(t)$, *for* $m = 1, ..., M$
3. $v_m = v_{m-1} - \lambda\int_{t_{m-1}}^{t_m}(v(t) - \overline{v})\,dt + \eta\int_{t_{m-1}}^{t_m}\sqrt{v(t)}dW_1(t)$, *for* $m = 1, ..., M$

Here $x_m = x(t_m)$ and $v_m = v(t_m)$. The full truncation scheme now approximates the integrals in algorithm 1. It is defined as follows:

$$y_m^{\Delta t} = y_{m-1}^{\Delta t} + \left(r - \frac{1}{2}\left(v_{m-1}^{\Delta t}\right)^+\right)\Delta t + \rho\sqrt{\left(v_{m-1}^{\Delta t}\right)^+}W_1(\Delta t) + \sqrt{(1-\rho^2)\cdot\left(v_{m-1}^{\Delta t}\right)^+}W_2(\Delta t),$$

$$v_m^{\Delta t} = v_{m-1}^{\Delta t} - \lambda\left(\left(v_{m-1}^{\Delta t}\right)^+ - \overline{v}\right)\Delta t + \eta\sqrt{\left(v_{m-1}^{\Delta t}\right)^+}W_1(\Delta t), \tag{4.3.5}$$

with initial conditions $y_0^{\Delta t} = x(t_0)$ and $v_0^{\Delta t} = v(t_0)$, and $\left(v_{m-1}^{\Delta t}\right)^+ = \max\left(v_{m-1}^{\Delta t}, 0\right)$. Here $y_m^{\Delta t}$ is the approximation of x_m and $v_m^{\Delta t}$ is the approximation of v_m.

4.4 The VG Model

The SDE of the VG process can be solved analytically. Recall that the dynamics of the VG process are

$$dx(t) = -\widetilde{\alpha}dt + dX(t;\sigma,\nu,\theta), \tag{4.4.1}$$

where $-\widetilde{\alpha}$ is the drift of the logarithmic price of the asset and the VG process $X(t;\sigma,\nu,\theta)$ is defined as

$$X(t;\sigma,\nu,\theta) = \theta\gamma(t;1,\nu) + \sigma W(\gamma(t;1,\nu)). \tag{4.4.2}$$

Here $\gamma(t;1,\nu)$ denotes a gamma process with mean 1 and variance ν. The analytic solution of (4.4.1) reads

$$\begin{aligned} x(T) &= x(t_0) - \widetilde{\alpha}(T-t_0) + X(T;\sigma,\nu,\theta) - X(t_0;\sigma,\nu,\theta) \\ &= x(t_0) - \widetilde{\alpha}(T-t_0) + \theta[\gamma(T;1,\nu) - \gamma(t_0;1,\nu)] + \sigma W(\gamma(T;1,\nu) - \gamma(t_0;1,\nu)) \\ &= x(t_0) - \widetilde{\alpha}(T-t_0) + \theta G(T-t_0,\nu) + \sigma W(G(T-t_0,\nu)). \end{aligned} \tag{4.4.3}$$

Here $G(T-t_0,\nu)$ is gamma distributed with mean $T-t_0$ and variance ν.

In the case we have to deal with path dependent options time is again discretized and we follow the same methodology as described above. So, if we consider an equidistant grid with $t_m = t_0 + m\Delta t$ for $m = 0, ..., M$, and $\Delta t = t_m - t_{m-1} = \frac{T-t_0}{M}$, then a path can be simulated as follows.

1. x_0 *is known*
2. $x_m = x_{m-1} - \alpha\Delta t + \theta G(\Delta t,\nu) + \sigma W(G(\Delta t,\nu))$

4.5 The Bates Model

In the Bates model we have a Poisson process that determines the jump times of the underlying asset. Whenever a jump occurs, the size of the jump is determined with a normally distributed random variable. In this section we describe an algorithm from Cont and Tankov (2004), which describes a way to simulate these jumps.

The Poisson jump process is described as follows:

$$dY(t) = j_{N(t)}dN(t), \tag{4.5.1}$$

where $N(t)$ is a Poisson process with intensity ξ, and $j_{N(t)}$ is the normally distributed jump size with mean μ_J and variance σ_J^2. The algorithm uses the fact that the number of jumps $N(T)$ of the Poisson process on the interval $[0,T]$ is a Poisson random variable with parameter λT. And, because of the memoryless property of the Poisson process, given the number of jumps $N(T)$, the exact moments of jumps on this interval are uniformly distributed, rearranged in increasing order.

The algorithm is described as

1. Simulate $N \sim poisson\,(\lambda T)$.
2. Simulate N independent random variables, $U_k \sim U\,[0,T]$
3. Rearrange U_k in increasing order.
4. Simulate $j_k \sim N\left(\mu_J, \sigma_J^2\right)$

The trajectory is given by

$$Y\,(t) = \sum_{i=1}^{N} \mathbb{I}_{U_i<t} \cdot j_i.$$

Chapter 5

Results of the Valuation Techniques

This chapter discusses the numerical results of the different pricing methodologies. Firstly, we focus on pricing of Bermudan and American options under the 1-D VG model and American options under the 2-D Heston model. For these derivatives the Fourier transform method and the P(I)DE method are considered. Secondly, we present the convergence of the Monte Carlo method for the pricing of European options. For this purpose we use the VG model, because paths can be simulated analytically with this model. Most computations in this chapter are done on a Intel(R) Pentium(R) 4, CPU 2.80 GHz. and 0.99 GB of RAM with Windows XP as operating system (for the Heston model we have used a different computer. The reason for this is clarified in that particular section).

5.1 Implementation Details

In this section we give some details on the implementation of the Fourier transform method. For implementation details of the PIDE for the variance gamma process we refer to Almendral and Oosterlee (2006) and of the PDE for the Heston model we refer to Ito and Toivanen (2006).

In the description of the Fourier transform method some variables were introduced, but need to be defined for the computations. Firstly, the damping factor α is introduced to ensure that the Fourier transform exists. Numerical testing showed that the impact of the damping factor is not significant. Therefore, we choose $\alpha = 0$. Other values of α may lead to a better convergence, but we do not have a good method to choose a available.

Secondly, the gridsize (or truncation bounds) need to be specified. We have considered gridsizes of $[-2, 2]$ and $[-8, 8]$. Research is done on choosing the truncation bounds, but a general description has not yet been found.

Furthermore, we mention that integrals are computed with the trapezoidal rule. Numerical tests showed that the performance of other numerical integration techniques, based on uniform grids, were less good. The use of kernel function does give slightly more accuracy, but the computational time is larger for this method. Nevertheless, the performance of both methods was approximately comparable.

Finally, reference values are calculated with the Fourier transform method on a grid $[-16, 16]$ with much more gridpoints than the highest amount of points in the particular tables.

5.2 Bermudan Options under VG

In this section a ten times exercisable Bermudan put option is priced. The reference value is: 9.040646114. The parameters for the VG model are chosen as:

$$S_0 = 100;\ K = 110;\ T = 1;\ r = 0.1;\ \sigma = 0.12;\ \theta = -0.14;\ \nu = 0.2$$

For the results of the PIDE method the code of Ariel Almendral, as described in Almendral and Oosterlee (2006), is used with a small adjustment for pricing Bermudans. Tables 5.2.1 and 5.2.3 present the results of the Fourier transform method described in section 2.1 to price the Bermudan option. The computational domain varies in these two tables. Results of the PDE approach can be found in the Tables 5.2.2 and 5.2.4.

N	Value	Error	CPU (sec.)
256	9.018266257	2.24E-02	0.01
512	9.038861805	1.78E-03	0.02
1024	9.042435904	1.79E-03	0.03
2048	9.041109546	4.63E-04	0.06
4096	9.040705950	5.98E-05	0.12
8192	9.040689325	4.32E-05	0.23
16384	9.040635923	1.02E-05	0.46
32768	9.040645035	1.08E-06	1.00

Table 5.2.1: A Bermudan option with the Fourier Transform Method: domain = [-8, 8]

The results in Table 5.2.1 are computed on a grid with boundaries $x_1 = -8$ and $x_N = 8$, where $x_i = \log\left(\frac{S_i}{K}\right)$. So, for the above settings the underlyings range from $110 \cdot e^{-8} \approx 0.0369$ to $110 \cdot e^{8} \approx 327905$. If we compare the results with the results in Table 5.2.3, we notice that a higher accuracy is obtained on a smaller domain, $x_1 = -2$ and $x_N = 2$ (i.e. $S_1 = 14.89$ and $S_N = 812.80$). In other words, a domain size of $[-8, 8]$ is too large. Higher accuracy is obtained, because with an equal amount of gridpoints the step size will be smaller.

N	M	Value	Error	CPU (sec.)
800	20	9.149501370	1.09E-01	0.62
1600	40	9.089208381	4.86E-02	0.78
3200	80	9.060377702	1.97E-02	1.94
6400	160	9.048742499	8.10E-03	3.46
12800	320	9.044333473	3.69E-03	10.67
25600	640	9.042408713	1.76E-03	73.26

Table 5.2.2: A Bermudan option with the PIDE: domain = [-8, 8]

In Table 5.2.2 we computed the option prices with the PIDE approach. The grid size is equal to the grid size used in Table 5.2.1, which enables us to compare accuracy and computational time. Obviously, the performance of the PIDE approach is worse than the Fourier transform method. A drawback of the PIDE approach for pricing Bermudans is that convergence is dependent on the number of time steps, while this is not the case for the Fourier transform method. In the case of the Fourier transform method the number of time steps is equal to the number of exercise moments, which is 10 in this example.

N	Value	Error	CPU (sec.)
256	9.042457644	1.81E-03	0.01
512	9.041103800	4.58E-04	0.02
1024	9.040702137	5.60E-05	0.03
2048	9.040689764	4.36E-05	0.06
4096	9.040635250	1.09E-05	0.11
8192	9.040644853	1.26E-06	0.23
16384	9.040646509	3.95E-07	0.46
32768	9.040646216	1.02E-07	1.00

Table 5.2.3: A Bermudan option with the Fourier Transform Method: domain = [-2, 2]

Table 5.2.4 presents the results of the PIDE approach where the domain is truncated at $x_1 = -2$ and $x_N = 2$. Notice that the values are exactly the same as in case of the PIDE approach with the larger domain, see Table 5.2.2. This is because the step sizes remained the same. Again we can conclude that truncation at $x_1 = -2$ and $x_N = 2$ is sufficient.

N	M	Value	Error	CPU (sec.)
200	20	9.149501370	1.09E-01	0.56
400	40	9.089208381	4.86E-02	0.60
800	80	9.060377702	1.97E-02	0.74
1600	160	9.048742499	8.10E-03	1.23
3200	320	9.044333473	3.69E-03	5.65
6400	640	9.042408713	1.76E-03	12.46

Table 5.2.4: A Bermudan option with the PIDE: domain = [-2, 2]

5.3 American Options under VG

In this section we price an American put option with the following VG parameters:

$$S_0 = 100;\ K = 90;\ T = 1;\ r = 0.1;\ \sigma = 0.12;\ \theta = -0.14;\ \nu = 0.2$$

The reference value is: 0.800820959. Again, for the PIDE method the code from Almendral and Oosterlee (2006) is used. In this section we compare the results of the Fourier transform method with and without Richardson extrapolation, and the results of the PIDE method. Results of the Fourier transform method without Richardson extrapolation are presented in Tables 5.3.1 and 5.3.2. Notice that for American options we need to increase the number of time steps to get convergence. This is because American options can be exercised continuously. With Richardson extrapolation, however, the number of time steps remains limited. The results of the Fourier transform method with Richardson extrapolation are presented in Table 5.3.3. Table 5.3.4 shows the results of the PIDE approach. In these tables M is the number of time steps.

N	M	value	error	CPU (sec.)
64	32	0.732801710	6.80E-02	0.01
128	64	0.943302862	1.42E-01	0.03
256	128	0.782298062	1.85E-02	0.09
512	256	0.815902589	1.51E-02	0.32
1024	512	0.800462640	3.58E-04	1.26
2048	1024	0.802368170	1.55E-03	5.30
4096	2048	0.801084859	2.64E-04	21.55

Table 5.3.1: An American option with the Fourier Transform Method without Richardson extrapolation: domain = [-8, 8]

Contrary to the Bermudan option results, the results from Table 5.3.1 show that the convergence of the American options is not smooth. This is because the solution converges from below in time direction and from above in space direction, see Figure 5.3.1. For a particular set of mesh and time steps, N and M, the solution may be very accurate because the error due to the time discretization cancels out by the error due to the space discretization.

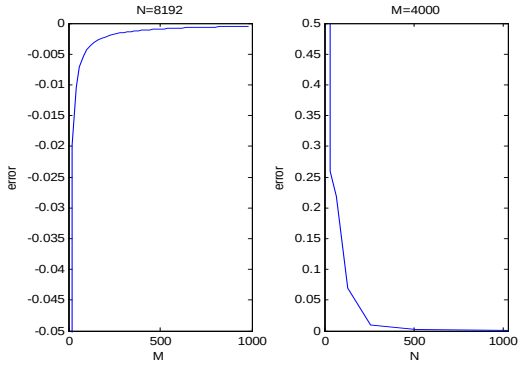

Figure 5.3.1: Left figure shows convergence in time direction, right figure convergence in space direction

In Table 5.3.2 we present results of the Fourier transform method on a smaller grid and with more gridpoints. As a result, the error of the time discretization is the leading error and we therefore observe smooth convergence.

In order to improve the performance of the Fourier transform method we apply Richardson extrapolation for the approximation of the American option prices. This technique enables us to keep the number of time steps low. Results are presented in Table 5.3.3. Notice that this method performs significantly better than the previously discussed methods. For the Richardson extrapolation, we used equation (2.4.11) with $k = 3$, which means that the American option is approximated by an $\widetilde{M}$ times exercisable Bermudan, a $2\widetilde{M}$ times exercisable Bermudan and a $4\widetilde{M}$ times exercisable Bermudan. Notice that at a certain point, for $N = 2048$, the error remains constant. This is because the error is due to the time discretization, i.e. due to the extrapolation.

N	M	value	error	CPU (sec.)
128	16	0.782367490	1.85E-02	0.01
256	32	0.786466343	1.44E-02	0.03
512	64	0.794954313	5.87E-03	0.09
1024	128	0.797418445	3.40E-03	0.32
2048	256	0.799174044	1.65E-03	1.25
4096	512	0.800011172	8.10E-04	5.27
8192	1024	0.800408156	4.13E-04	21.45

Table 5.3.2: An American option with the Fourier Transform Method without Richardson extrapolation: domain = [-2, 2]

N	$\widetilde{M}$	value	error	CPU (sec.)
128	16	0.828339573	2.75E-02	0.05
256	16	0.800898955	7.80E-05	0.09
512	16	0.802828594	2.01E-03	0.17
1024	16	0.800549352	2.72E-04	0.31
2048	16	0.800806096	1.49E-05	0.66
4096	16	0.800803620	1.73E-05	1.29
8192	16	0.800800229	2.07E-05	2.61

Table 5.3.3: An American option with the Fourier Transform Method with Richardson extrapolation: domain = [-2, 2]

Finally, we present the results of the PIDE approach for American options under the VG model in Table 5.3.4. Notice that the Fourier transform method outperforms the PIDE approach, though, the differences are not as large as in case of the Bermudan options.

N	M	Value	Error	CPU (sec.)
200	40	0.765903160	3.49E-02	0.57
400	80	0.800444986	3.76E-04	0.61
800	160	0.806035551	5.21E-03	0.86
1600	320	0.804953525	4.13E-03	1.75
3200	640	0.803292323	2.47E-03	10.61
6400	1280	0.802161616	1.34E-03	24.27

Table 5.3.4: An American option with the PIDE: domain = [-2, 2]

5.4 American Options under Heston

The valuation of derivatives under the Heston model is more complicated than for one-dimensional models, such as the VG model. Not only the theory becomes more complicated, but also the computational time to price an option increases significantly. This section compares the already extensively studied PDE approach to the novel Fourier transform technique, described in Chapter 2. For the PDE results we have used the fast and accurate implementation of Ito and Toivanen (2006). Their code is written in Fortran 90, which is not supported by the computer used for all other computations. Therefore, the computations in this section are done on a different computer, namely an Intel(R) Pentium(R) 4, CPU 3.00 GHz. and 2 GB of RAM in a Linux (kernel

2.6.16.7) environment. Tables 5.4.1 and 5.4.2 present these results. The results are obtained with the following settings:

$$S_0 = 10;\ K = 10;\ T = 0.25;\ r = 0.1;\ \lambda = 5;\ \eta = 0.9;\ \overline{v} = 0.16;\ v_0 = 0.25;\ \rho = 0.1$$

N	J	$\widetilde{M}$	Value	Error	CPU (sec.)
16	10	1	26.085209768	2.53E+01	1.02
32	20	1	0.859509201	6.35E-02	4.58
64	30	1	0.791900915	4.08E-03	13.68
128	40	1	0.794983081	9.94E-04	34.22
256	50	1	0.795989393	1.24E-05	85.16

Table 5.4.1: An American option with the Fourier transform method: x-domain = [-2, 2] and v-domain = [0,2]

For the Fourier transform method we have used the repeated Richardson extrapolation with $k = 4$ in equation (2.4.11). In other words, the American option is approximated by $\widetilde{M}$, $2\widetilde{M}$, $4\widetilde{M}$ and $8\widetilde{M}$ exercisable Bermudan options.

N	J	M	Value	Error	CPU (sec.)
17	9	8	0.788562383	7.41E-03	0.00
33	17	16	0.794060919	1.92E-03	0.02
65	33	32	0.795587537	3.89E-04	0.17
129	65	64	0.795937587	3.94E-05	1.87
257	129	128	0.795963450	1.36E-05	17.70

Table 5.4.2: An American option with the PIDE: S-domain = [0, 20] and v-domain = [0, 1]

If we compare the results of the Fourier transform method with Richardson extrapolation to the results of the two-dimensional PDE approach, we may conclude that under this model the Fourier transform method does not perform best. However, the implementations are done in different programming languages. The Fourier transform method is implemented in Matlab and the PDE is implemented in Fortran 90. We compared the speed of both languages by implementing a program computing the trapezoidal rule in both languages and ran it a thousand times. It turned out that for this particular example the Fortran 90 implementation was more than 10 times faster than Matlab. We therefore realize that a true comparison cannot be made in this way. What we can infer from the results is that the novel Fourier transform method converges to the correct solution.

5.5 Convergence of the Monte Carlo Method

As stated above, the Monte Carlo simulation method is applicable for many different payoff functions and different pricing models. However, we do not recommend to use this Monte Carlo simulation if another method is available for a specific option and asset price model. This is because the convergence of this method is extremely slow and not smooth. Therefore, Monte Carlo simulation techniques are only useful for highly path-dependent options or multi-dimensional models. Figure 5.5.1 presents the convergence of a European option with the Monte Carlo method as a function of CPU-time. This plot shows the non-smoothness of the convergence, but most importantly it gives an indication of the large variance even after two seconds.

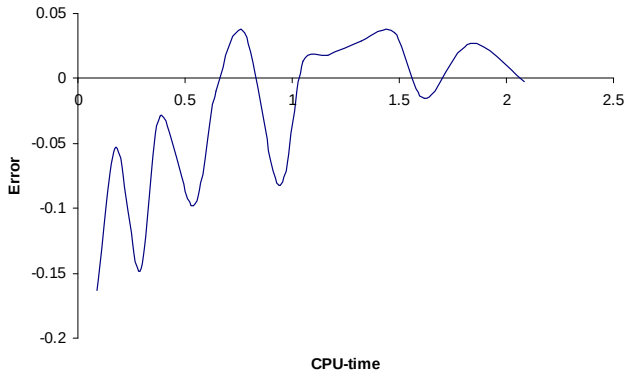

Figure 5.5.1: Convergence of a European option with the Monte Carlo method

The large variance can be reduced with variance reduction techniques, e.g. antithetic variates. Though, it will never perform as well as the Fourier transform method. By, comparing the Monte Carlo method with the Fourier transform method, presented in Figure 5.5.2, we notice that the Fourier transfrom method convergences smoothly and about $1,000,000$ times faster. Therefore, we recommend to use the Monte Carlo method only if other methods cannot be used, i.e. for highly path-dependent payoffs or multi-dimensional models.

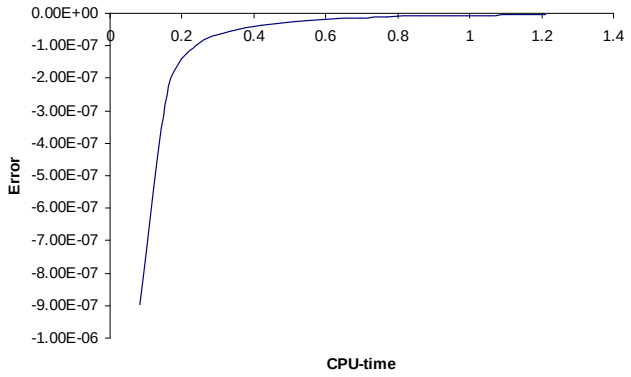

Figure 5.5.2: Convergence of a European option with the Fourier transform method

Chapter 6

Conclusions and Recommendations for the Valuation Techniques

In Part II of this thesis we have presented various valuation techniques for pricing callable (early exercise) products under different models. Here we concentrated on comparing a novel quadrature technique based on the FFT with the already extensively studied P(I)DE approach. This novel quadrature techniques is based on the fact that for one-dimensional exponential Lévy models the probability density is not dependent on the current state of the underlying, but only on the change of the underlying. It turns out that due to this fact the quadrature formula can be recognized as a convolution. The methodology was extended for the Heston model, which is two-dimensional. The fact that the probability density is not dependent on the current state of the underlying, but only on the change of the underlying, does not hold for the variance dimension. This introduced an extra integral for the computation of early exercise options.

In Chapter 5 we have investigated the performance of the quadrature technique by comparing it to implementations of P(I)DEs. Firstly, we compared the valuation of Bermudan options. In this setting we have chosen to use the one-dimensional variance gamma model. The quadrature technique outperformed the PIDE method. Next, we studied the valuation of American options under the variance gamma model. For the American options the quadrate method also performs significantly better than the PIDE method. From these results we can conclude that the quadrature technique is better suited to value callable options under one-dimensional models. However, a drawback of the quadrature approach is that it is not applicable to models for which the characteristic function is not known, e.g. the local volatility model. A drawbacks of the PIDE approach is that we have to deal with a time direction, which means that much more time steps are necessary than the number of exercise moments.

Secondly, we compared the valuation of American options under the two-dimensional Heston model. It turned out that we cannot conclude whether the PDE or the quadrate technique performs best, because the codes were written in different programming languages. We can conclude that the novel quadrature approach for the Heston model converges to the correct solution and is also promising of this model.

Finally, we showed that the Monte Carlo method should only be used if other methods are not applicable. This is because of the non-smooth and slow convergence of the Monte Carlo method.

We finalize this chapter with some suggestions for further research:

- The comparison of the quadrature technique and the PDE in the Heston model is done in

different programming languages. As a result, we cannot conclude which method performs best. We suggest to investigate the performance of both methods in the same programming language.

- Numerical testing showed that the Richardson extrapolation technique is a very satisfactory method for obtaining high accuracy for the valuation of American options with a limited number of exercise moments. Richardson extrapolation is applicable to almost any kind of numerical approximation. Therefore, we expect promising results when applying this technique to the variance discretization in the Heston model.

- Especially for one-dimensional models the quadrature method has proven to be very accurate and fast. Though, we are not yet really able to say anything in advance about the error. In case of P(I)DEs more research is done to this aspect and it is interesting to get a grip on resulting errors for the quadrature method as well.

- In this thesis we considered the valuation of early exercise options. Adapted Fourier transform techniques are also applicable to other payoffs. We suggest to study the performs of this method for these different payoffs as well.

- Finally, we suggest to study on a way to determine the truncation bounds of the integral in the quadrature method. We noticed that for our example $L = -2$ and $U = 2$ was large enough, but we are not able to give exact bounds for any kind of option.

Part III

Model Risk

Chapter 7

Exotic Options

European and American options are commonly traded on several markets in the world. Therefore, market prices are available for European and American options. Options that are not traded on official markets are known as exotic options. Usually they are traded between banks, insurance companies, etc or used as bonusses for the managing moard. Market prices are therefore typically not known for these products. This section describes the features of some popular exotic derivatives and ends with the description of two recent trades by Rabobank International.

7.1 Vanilla Arithmetic Cliquets

A cliquet is a product which is dependent on the returns between two observation dates. The return G_i of an underlying $S(t)$ is defined as

$$G_i = \frac{S_i - S_{i-1}}{S_{i-1}}, \tag{7.1.1}$$

where S_i is the closing price of the underlying at the i-th observation date. The returns of a cliquet, U_i, are bounded by a local floor (F_i) for the losses and a local cap (C_i) for the gains:

$$U_i = \min\left(\max\left(F_i, G_i\right), C_i\right). \tag{7.1.2}$$

The final payoff of the arithmetic cliquet, subject to a global floor (F), and cap (C), reads

$$V(T) = N \cdot \min\left(\max\left(\sum_i U_i, F\right), C\right), \tag{7.1.3}$$

where N is the notional. The notional is the amount that is invested in the product.

Many variants of these cliquets exist, such as reverse cliquets, cancellable cliquets, swing cliquets, cliquets with lock-ins and combinations of these. They are not treated in this thesis.

Example 1 *Consider a* 3 *year arithmetic cliquet with monthly observations given by the end dates of the month and a notional amount of* 100. *The periodic returns are locally floored at* -10% *and capped at* 15%. *Subject to a global floor of* -2.5 *and a global cap of* 4.5 *the arithmetic cliquet pays at maturity:*

$$V(T) = 100 \cdot \min\left(\max\left(\sum_{i=1}^{36} U_i, -2.5\right), 4.5\right),$$

where $U_i = \min\left(\max\left(-10\%, G_i\right), 15\%\right)$.

7.1.1 A Fourier Pricing Methodology

An adapted Fourier pricing methodology is applicable for the pricing of arithmetic cliquets. This methodology is described in this section and applies to all models for which the characteristic function is known. The method originates from Den Iseger and Oldenkamp (2005). Only for the local volatility model the Monte Carlo pricing technique must be used.

The price of a cliquet is calculated as the discounted expected payoff. The payoff of a cliquet reads

$$V(T) = N \min(\max(Z, F), C), \tag{7.1.4}$$

where $Z = \sum_{i-1}^{n} \min(\max(F_i, R_i - 1), C_i)$, with $R_i = \frac{S_i}{S_{i-1}}$. If the density of Z is known, denoted by $p(x)$, then the price of a cliquet equals

$$\begin{aligned} V(t_0) &= e^{-r(T-t_0)} \mathbb{E}_{t_0} [N \min(\max(Z, F), C)] \\ &= e^{-r(T-t_0)} N \left(\int_F^C x p(x)\, dx + F \int_{-\infty}^F p(x)\, dx + C \int_C^\infty p(x)\, dx \right). \end{aligned} \tag{7.1.5}$$

A standard numerical integration technique, e.g. the trapezoidal rule, is used to evaluate the integrals. The problem in (7.1.5) is that for most models the density of Z is not known analytically. However, we know the characteristic function of the logarithmic returns and the inverse Fourier transform is used to obtain the density of the logarithmic returns:

$$p(z) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \mathrm{Re}\left(e^{-i\omega z} \widehat{p}(\omega)\right) d\omega.$$

Here the integral is calculated numerically, as described in Section 2.3.1. If we denote by f_{z_i}, $i = 1, ..., M$, the density of the logarithmic return at time t_i, $U_i = \min(\max(F_i, R_i - 1), C_i)$ and f_{U_i} as the density of U_i, then the following identities hold:

$$\begin{array}{ll} f_{U_i}(x) = 0, & \text{for } x < F_i \\ \mathbb{P}(U_i = F_i) = F_{z_i}(\log(F_i + 1)), & \\ f_{U_i}(x) = \frac{f_{z_i}(\log(x+1))}{x+1}, & \text{for } F_i < x < C_i \\ \mathbb{P}(U_i = C_i) = 1 - F_{z_i}(\log(C_i + 1)), & \\ f_{U_i}(x) = 0, & \text{for } C_i < x. \end{array} \tag{7.1.6}$$

Hence, the Fourier transform is given by

$$\widehat{f}_{U_i}(\omega) = e^{i\omega F_i} F_{z_i}(\log(F_i + 1)) + \int_{F_i}^{C_i} e^{i\omega x} \frac{f_{z_i}(\log(x+1))}{x+1} dx + e^{i\omega C_i} (1 - F_{z_i}(\log(C_i + 1))). \tag{7.1.7}$$

Now, the characteristic function of Z can be found as follows:

$$\widehat{g}(\omega) = \mathbb{E}\left[e^{i\omega Z}\right] = \mathbb{E}\left[e^{i\omega \sum_{i-1}^{n} U_i}\right] = \mathbb{E}\left[e^{i\omega U_1}\right] \cdots \mathbb{E}\left[e^{i\omega U_n}\right] = \widehat{f}_{U_1}(\omega) \cdots \widehat{f}_{U_n}(\omega). \tag{7.1.8}$$

Finally, the inverse Fourier transform is used to obtain the density of Z:

$$g(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \mathrm{Re}\left(e^{-i\omega x}\widehat{g}(\omega)\right) d\omega. \tag{7.1.9}$$

Again, this integral is approximated by the methodology described in Section 2.3.1.

7.2 Discrete Knock-out Barrier Options

The second exotic option we consider is a so-called discrete knock-out barrier option. This is a regular European option with strike price K and maturity T, which ceases to exist if the asset price reaches a certain barrier level, H, at one of the prespecified observation dates, t_i. When the barrier is greater than the initial asset price, the option is referred to as an up-and-out option; when the barrier is less than the initial asset price, it is referred to as a down-and-out option. If the barrier is hit the options pays some rebate Rb at the option's maturity. So, the payoff at maturity of a discrete knock-out barrier option reads

- Up-and-out:

$$V(T) = \left(\max\left(\varphi\left(S(T) - K\right), 0\right) - Rb\right) \mathbb{I}_{[\max_i S(t_i) < H]} + Rb \tag{7.2.1}$$

- Down-and-out:

$$V(T) = \left(\max\left(\varphi\left(S(T) - K\right), 0\right) - Rb\right) \mathbb{I}_{[\min_i S(t_i) > H]} + Rb \tag{7.2.2}$$

where $\mathbb{I}_A$ is the indicator function defined as $\mathbb{I}_A(x) = \begin{cases} 1 & x \in A \\ 0 & x \notin A \end{cases}$ and φ is defined in Section 0.1.

Example 1 *Consider a discrete up-and-out call option with strike price* 100*, maturity in* 1 *year, a barrier level of* 140 *and monthly observations. If the option ceases to exist no rebate is paid. Then the payoff at maturity equals:*

$$\max\left(S(T) - 100, 0\right) \mathbb{I}_{\left[\max_{1 \le i \le 12} S(t_i) < 140\right]}$$

7.2.1 Pricing Methodology

The pricing of discrete barrier options is similar to pricing Bermudan options. As described in Section 2.1.2, a Bermudan option equals its payoff if the corresponding European price is lower than the payoff at the prespecified exercise dates. For a discrete barrier option, the price equals the discounted rebate if the underlying is above (up-and-out) or below (down-and-out) the barrier level.

Denote the set of observation dates as $\mathcal{T} = \{t_0, ..., t_M\}$, $t_0 < t_1 < \cdots < t_{M-1} < t_M = T$. If we denote $C(x, t_k)$ as the continuation value at $t_k \in \mathcal{T}$ and $U(x, t_k)$ the up-and-out option value, then we have for $k = 0, ..., M$, with $C(x, t_M) = g(x)$ the exercise payoff,

$$U(x, t_k) = \begin{cases} e^{-r(T-t_k)} Rb & \text{, for } x \ge \log\left(\frac{H}{K}\right) \\ C(x, t_k) & \text{, for } x < \log\left(\frac{H}{K}\right) \end{cases}, \tag{7.2.3}$$

$$C(x, t_k) = e^{-r(t_{k+1}-t_k)} \mathbb{E}\left[U(x, t_{k+1})\right]. \tag{7.2.4}$$

Here $x = \log\left(\frac{S}{K}\right)$ the logarithmic stock price.

Similarly, if we denote by $D(x, t_k)$ the down-and-out option value at time $t_k \in \mathcal{T}$, we have for $k = 0, ..., M$

$$D(x, t_k) = \begin{cases} e^{-r(T-t_k)} Rb & \text{, for } x \leq \log\left(\frac{H}{K}\right) \\ C(x, t_k) & \text{, for } x > \log\left(\frac{H}{K}\right) \end{cases}, \tag{7.2.5}$$

$$C(x, t_k) = e^{-r(t_{k+1}-t_k)} \mathbb{E}\left[U(x, t_{k+1})\right]. \tag{7.2.6}$$

Obviously, the payoff of a barrier option has a discontinuity at the (logarithmic) barrier level. This causes some numerical troubles, which are undesirable. Firstly, the solution approaches the true solution in a sawtooth fashion. Secondly, the price converges very slowly as the number of grid points increases.

Smooth convergence can be obtained by placing one of the grid points on the logarithmic barrier level. We use a computationl grid as follows:

$$x_i = -\frac{N}{2}\Delta x + h + (i-1)\Delta x. \tag{7.2.7}$$

At $x_{\frac{N}{2}+1} = h$ the option value equals the rebate, e.g. $Rb = 0$. However, including this in the algorithm gives us an option value that is consequently underestimated. This is because the surface below the option value between $x_{\frac{N}{2}}$ and h is not taken into account in the integration, which causes slow convergence. Similarly, choosing this value equal to the European price means that the price is overestimated. To enhance the speed of convergence we set the option value at h as the interpolation between the surrounding grid points. Linear interpolation increases the speed of convergence significantly, but testing shows that cubic interpolation performs even better. The idea of this interpolation technique was already applied in trees by Derman, Kani, Ergener and Bardhan (1995).

For the LV model we employ the Monte Carlo simulation approach. For all other models, the Fourier transform method is applicable. We may treat the barrier option as a set of M European options, each with maturity Δt and a payoff described by (7.2.3) for an up-and-out and (7.2.5) for a down-and-out barrier option. Notice that the option prices should be resolved iteratively, because these payoffs are not known in advance.

7.3 Napoleon Options

The important factors in the payoff of a Napoleon option are a known fixed coupon and the worst returns of an index or an equity over specified time periods. A general Napoleon option has multiple payoff coupons and each of the coupons consist of multiple so-called performances. The performances can be capped or floored as well as the coupons. Adding a floor to a performance or a coupon will rease the price of a Napoleon option, and the effect of adding a cap to a performance or a coupon will bring the opposite. By adjusting the caps and the floors an investor can balance between the price and the amount of risk he is willing to take. The performance P (without a cap or a floor) is calculated as

$$P = r_0 + r \cdot \min(G_i), \tag{7.3.1}$$

where r_0 is a fixed coupon, r is a scaling factor and G_i is as in (7.1.1). Suppose a coupon consists of M performances P_m, $m = 1, ..., M$. Then a coupon C (without a cap or a floor) is defined as

$$C = \sum_{m=1}^{M} P_m. \tag{7.3.2}$$

The cash flow amount equals the notional times the coupon: $N \cdot C$.

Example 1 *Consider a* 4 *year Napoleon option with annual coupons, semi-annual performances and monthly returns. The notional equals* 100*, the Napoleon has a fixed rate of* 10%*, there is no scaling and the product is without any caps or floors. Each coupon is then calculated as*

$$C_k = P_1^k + P_2^k, \quad k = 1, 2, 3, 4$$

where the performances P_1^k *and* P_2^k *are calculated as*

$$P_1^k = 10\% + \min_{i=1,..,6} \left(G_i^k\right), \quad P_2^k = 10\% + \min_{i=7,..,12} \left(G_i^k\right).$$

Here G_i^k *is the return of the* i*-th month of the* k*-th year.*

The payoff of a Napoleon option is too difficult to fit into the PDE or FTM framework. Therefore, this option is priced with the Monte Carlo method.

7.4 Rabo Top Europe Note and Rabo Deep Autocall Certificate

The Rabo Top Europe Note is a recently issued structured product by Rabobank International. The underlying index of this product is the Dow Jones EUROSTOXX 50 Index. The note may be seen as a combination of a lookback option and a barrier option. A detailed overview of the characteristics of this product are presented in Appendix C. The option is priced by the Monte Carlo method.

Another structured product which is recently issued by Rabobank International is the Rabo Deep Autocall Certificate. This product guarantees a fixed redemption if the underlying reaches a certain barrier at one of the prespecified observation dates. In that case, the product ceases to exist. Otherwise, redemption is determined using the performance of the underlying. A detailed overview of the characteristics of this product are presented in Appendix D. The option is priced by the Monte Carlo method.

7.5 A Monte Carlo Pricing Methodology

The Monte Carlo Pricing Methodology can be applied to all models. The main drawback of Monte Carlo techniques is its slow convergence, see section 5.5. Therefore, this technique will only be applied in case of the Local Volatility (LV) model, for which the characteristic function is not known, or when the product cannot be priced in another framework. Consider a uniform time discretization $t_0 < t_1 < \cdots < t_N = T$, where t_0 is the current date and T the maturity of the contract. Furthermore, consider the observation times $t_1^{obs} < t_2^{obs} < \cdots < t_M^{obs} \leq T$, where

M is the number of observation dates. In general M is much smaller than N, $M \ll N$. In order to calculate the returns we need to store the previous observation date, e.g. to calculate the k-th return the $(k-1)$-th observation value needs to be stored. As soon as the k-th observation date is known, the return can be calculated and the k-th observation date becomes the previous observation date. Observation dates can be between two time points, e.g. $t_i < t_k^{obs} < t_{i+1}$. In this case, the underlying value is calculated with linear interpolation.

Chapter 8

Calibration

So far, we have presented pricing methodologies for models with various parameters and exotic options with sometimes complicated payoffs. The pricing methods assume that the model parameters are known. The way in which these parameters are usually calculated is presented in this section. The idea is to find those parameters for which the difference between the vanilla option model prices and market prices is minimized. This process is known as the calibration process. Calibration of the local volatility model is done different from the other models and will be treated separately. The models are calibrated to the Dow Jones EUROSTOXX 50 volatility surface of 30 May 2006.

In order to obtain satisfactory results for inter-model risk, we calibrate a benchmark model to the market and all other models to the benchmark model. This is because the calibrated benchmark model, by definition, will have a smooth volatility surface. This will lead to small calibration errors for all other models. This procedure is common in the financial industry. Our benchmark model will be Bates model, because it is most likely to fit the market data best, as it contains the largest number of parameters to calibate.

8.1 The Non-Local Volatility Models

The calibration process appears to be at first sight a straightforward process of minimizing an objective function, which measures the difference between market prices and model prices. In practice this process is not as straightforward as expected. Firstly, different objective functions result in different optimal values for the parameters. Secondly, the objective functions have several local minima. To find the global minimum one should apply stochastic optimization techniques, e.g. genetic algorithms. This does not fit in the scope of this thesis. Here, we choose to use a heuristic technique of choosing smart starting parameters and apply local optimization techniques for calibration. With respect to the objective function, Ng (2005) concluded that the so called Average Absolute Error (AAE) is the best objective function to use:

$$AAE = \frac{1}{N}\sum_{i=1}^{N}\left|C^i_{\text{model}}(\theta) - C^i_{\text{market}}\right|, \tag{8.1.1}$$

where θ denotes the set of model parameters and C_i represents the call prices. Therefore, we also use this objective function in the following chapters.

For the calibration of the Bates model we find several local minima with the minima close to each other. These parameters are presented in Table 8.1.1.

Set	λ	η	$\overline{v}$	v_0	ρ	ξ	μ_j	σ_j	AAE
1	0.108	0.170	0.163	0.024	-1.000	0.345	0.110	0.000	4.395
2	0.147	0.191	0.137	0.026	-0.905	4.329	0.016	0.018	4.831
3	0.142	0.189	0.139	0.026	-0.906	4.329	0.017	0.018	4.747
4	0.153	0.193	0.135	0.026	-0.912	4.425	0.016	0.018	4.964
5	0.143	0.189	0.138	0.026	-0.906	4.329	0.017	0.018	4.765

Table 8.1.1: Different calibrated Bates parameters with calibration error

The first set of parameters in this table has a significantly lower objective function value than the other sets. However, this set will not be used as our benchmark, because the parameters are not very realistic. The correlation, ρ, indicates that the Brownian motion of the stock price and the Brownian motion of the variance are perfectly negatively correlated, which is unrealistic. Another issue is that the variance of the jump size gets zero. This means that whenever a jump occurs it is always a jump of the same size, which is also unrealistic. The impact of the different sets of parameters on the exotic option prices is relatively small. This is studied in detail in Section 8.3. We choose the smallest objective function with a realistic set of parameters, which is set 3, as our benchmark model.

The Black-Scholes model, the Heston model and the Variance Gamma model are calibrated to this benchmark set. Again, several local minima are found, and we will use the lowest AAE. The resulting parameters are presented in Table 8.1.2. Notice that for the BS model as well as for the VG mdoel the AAE error is still significant.

Black-Scholes $\sigma = 0.214$	$AAE = 50.653$
Heston $\lambda = 0.136$, $\eta = 0.190$, $\overline{v} = 0.144$, $v_0 = 0.029$, $\rho = -0.834$	$AAE = 0.545$
Variance Gamma $\nu = 5.139$, $\theta = -0.116$, $\sigma = 0.193$	$AAE = 18.615$

Table 8.1.2: Calibration results

8.2 The Local Volatility Model

To calibrate the local volatility model we need to find an explicit expression for the local volatility function. The local volatility function reads, see equation (1.5.2),

$$\sigma(x,t)^2 = 2\frac{\frac{\partial}{\partial T}C(x,x,t) + \mu\frac{\partial}{\partial k}C(x,x,t) + (r-\mu)C(x,x,t)}{\frac{\partial^2}{\partial k^2}C(x,x,t) - \frac{\partial}{\partial k}C(x,x,t)}. \tag{8.2.1}$$

So, we need to know the derivatives of the call prices, but we only know these call prices at fixed points. In order tofind these derivatives we define an interpolation method to determine the market prices for a complete price surface. Within Rabobank International the interpolation is

done on the implied volatilities with a cubic spline interpolation in the strike direction and linear interpolation in the time direction. With this surface we are able to approximate the derivatives as follows:

$$\begin{aligned}
\frac{\partial}{\partial T}C(x,x,t) &= \frac{C(x,x,t+\Delta T)-C(x,x,t)}{\Delta T},\\
\frac{\partial}{\partial k}C(x,x,t) &= \frac{C(x,x+\Delta k,t)-C(x,x-\Delta k,t)}{2\Delta k},\\
\frac{\partial^2}{\partial k^2}C(x,x,t) &= \frac{C(x,x+\Delta k,t)-2C(x,x,t)+C(x,x-\Delta k,t)}{\Delta k^2}.
\end{aligned}$$

Further, we assume no dividends in this calibration. Therefore, $\mu = r$ and (8.2.1) can be simplified to

$$\sigma(x,t)^2 = 2\frac{\frac{\partial}{\partial T}C(x,x,t)+r\frac{\partial}{\partial k}C(x,x,t)}{\frac{\partial^2}{\partial k^2}C(x,x,t)-\frac{\partial}{\partial k}C(x,x,t)}. \tag{8.2.2}$$

8.3 Results for Model Risk

For several exotic products we compare the value and delta for the different parameter sets. For Bates model we will study the intra-model risk. For all other models we only examine the differences between the models (inter-model risk). The products that are examined are defined in the following tables. In Table 8.3.1 different cliquets (Section 7.1) are presented. We define some abbreviations to safe space in the tables:

sa: semi-annually , q: quarterly , Perf: performances,
LF: local floor , LC: local cap , Mat: Maturity

Name	Mat	Frequency	LF	LC	Floor	Cap
Cliquet1	$T=1$	sa	$F_i=-0.1$	$C_i=0.1$	$F=0$	$C=0.15$
Cliquet2	$T=3$	sa	$F_i=-0.1$	$C_i=0.1$	$F=0$	$C=0.5$
Cliquet3	$T=1$	sa	$F_i=-0.2+0.05i$	$F_i=0.05i$	$F=0$	$C=10^{10}$
Cliquet4	$T=3$	sa	$F_i=-0.2+0.05i$	$F_i=0.05i$	$F=0$	$C=10^{10}$
Cliquet5	$T=1$	monthly	$F_i=-10^9$	$C_i=0.03$	$F=0$	$C=0.3$
Cliquet6	$T=3$	monthly	$F_i=-10^9$	$C_i=0.03$	$F=0$	$C=0.9$

Table 8.3.1: Cliquet settings, Notional = 100

In Table 8.3.2 the essential features of the barrier option (Section 7.2) are described.

Name	Strike	Mat	Frequency	C/P	Barrier	U/D	Rebate
UC1	$K = S_0$	$T = 1$	sa	Call	$H = 1.3 \cdot S_0$	Up	$R = 0$
UC2	$K = 0.9 \cdot S_0$	$T = 1$	sa	Call	$H = 1.1 \cdot S_0$	Up	$R = 0$
UC3	$K = S_0$	$T = 1$	monthly	Call	$H = 1.3 \cdot S_0$	Up	$R = 0$
UC4	$K = 0.9 \cdot S_0$	$T = 1$	monthly	Call	$H = 1.1 \cdot S_0$	Up	$R = 0$
UC5	$K = S_0$	$T = 3$	annually	Call	$H = 1.3 \cdot S_0$	Up	$R = 5$
UC6	$K = 0.9 \cdot S_0$	$T = 3$	annually	Call	$H = 1.1 \cdot S_0$	Up	$R = 5$
DP1	$K = S_0$	$T = 1$	sa	Put	$H = 0.8 \cdot S_0$	Down	$R = 0$
DP2	$K = 1.1 \cdot S_0$	$T = 1$	sa	Put	$H = 0.9 \cdot S_0$	Down	$R = 0$
DP3	$K = S_0$	$T = 1$	monthly	Put	$H = 0.8 \cdot S_0$	Down	$R = 0$
DP4	$K = 1.1 \cdot S_0$	$T = 1$	monthly	Put	$H = 0.9 \cdot S_0$	Down	$R = 0$
DP5	$K = S_0$	$T = 3$	annually	Put	$H = 0.8 \cdot S_0$	Down	$R = 5$
DP6	$K = 1.1 \cdot S_0$	$T = 3$	annually	Put	$H = 0.9 \cdot S_0$	Down	$R = 5$

Table 8.3.2: Knock-out barrier settings, $S_0 = 100$

Name	Mat	Coupons	Perf	Returns	LF	LC	Floor	Cap
Nap1	$T = 2$	annually	sa	monthly	F_i= 0.02	C_i= 0.1	$F = 0$	$C = 10^{10}$
Nap2	$T = 4$	annually	sa	monthly	F_i= 0.02	C_i= 0.1	$F = 0$	$C = 10^{10}$
Nap3	$T = 2$	annually	sa	q	F_i= 0.02	C_i= 0.1	$F = 0$	$C = 10^{10}$
Nap4	$T = 2$	annually	q	monthly	F_i= 0.02	C_i= 0.1	$F = 0$	$C = 10^{10}$
Nap5	$T = 2$	sa	q	monthly	F_i= 0.02	C_i= 0.1	$F = 0$	$C = 10^{10}$
Nap6	$T = 2$	annually	sa	monthly	F_i= 0.02	C_i= 0.1	$F = 0.05$	$C = 0.15$
Nap7	$T = 2$	annually	sa	monthly	F_i= 0.01i	C_i= 0.05i	$F = 0$	$C = 10^{10}$

Table 8.3.3: Napoleon settings, notional = 100, fixed rate = 8% and scaling =1

And Table 8.3.3 presents several Napoleon options (Section 7.3).
The delta is approximated numerically:

$$\Delta = \frac{\partial V}{\partial S} = \frac{V(S+h,T) - V(S-h,T)}{2h}, \tag{8.3.1}$$

where $h = 0.1$. In the computations we assume a risk-free interest rate of 5%, i.e. $r = 0.05$. The maximum differences are calculated as relative difference between the largest and smallest price:

$$\text{max. diff.} = \frac{\max(V_i) - \min(V_i)}{\max(V_i)}$$

8.4 Intra-Model Risk

Intra-model risk is defined as the maximum price difference within one model for a certain derivative. Here the intra model risk for Bates model is approximated by looking at the maximum price difference between the five parameter sets as specified in Table 8.1.1. We will consider the derivatives that are described in the beginning of this chapter.

For all products we only notice small price differences. Especially the Napoleon option does not seem to be really dependent on the set of parameters.

The Barrier option is the most sensitive product to the parameter sets. Though, the price differences are still limited, but the maximum difference in the delta is considerable. If we take

Set	1	2	3	4	5	max. diff.
Cliquet1	5.50	5.58	5.57	5.59	5.57	1.61%
Cliquet2	12.27	12.59	12.54	12.62	12.54	2.77%
Cliquet3	4.75	4.83	4.82	4.84	4.82	1.86%
Cliquet4	27.33	27.85	27.83	27.88	27.83	1.97%
Cliquet5	2.53	2.60	2.58	2.60	2.58	2.69%
Cliquet6	2.28	2.32	2.29	2.32	2.29	1.72%

Table 8.4.1: Cliquet option prices under Bates

a closer look at were the differences come from, we notice that for all products the results of the first set differs a lot from the other sets. As discussed before, this set is unrealistic and should not be used because of that reason. If we only consider the remaining sets all differences in the prices and deltas are even smaller.

Set	1		2		3		4		5		max. diff.	
	V	Δ	V	Δ	V	Δ	V	Δ	V	Δ	V	Δ
UC1	3.26	0.17	3.20	0.16	3.21	0.16	3.19	0.16	3.21	0.16	2.15%	5.81%
UC2	2.77	0.05	2.67	0.04	2.67	0.04	2.66	0.04	2.67	0.04	3.97%	4.35%
UC3	3.43	0.19	3.34	0.18	3.34	0.18	3.33	0.18	3.34	0.18	2.92%	7.89%
UC4	2.15	−0.06	2.03	−0.06	2.03	−0.06	2.03	−0.06	2.03	−0.06	5.58%	8.06%
UC5	2.50	0.07	2.48	0.06	2.48	0.06	2.48	0.06	2.48	0.06	0.80%	4.55%
UC6	2.60	0.06	2.58	0.06	2.59	0.06	2.58	0.06	2.59	0.06	0.77%	1.59%
DP1	4.37	−0.10	4.42	−0.08	4.40	−0.08	4.42	−0.08	4.40	−0.08	1.13%	24.75%
DP2	3.57	0.14	3.46	0.14	3.45	0.14	3.44	0.14	3.45	0.14	3.64%	1.46%
DP3	3.32	−0.07	3.35	−0.06	3.34	−0.06	3.35	−0.06	3.34	−0.06	0.90%	18.57%
DP4	2.59	0.12	2.48	0.13	2.47	0.13	2.48	0.13	2.47	0.13	4.63%	6.20%
DP5	3.87	−0.04	3.86	−0.04	3.85	−0.04	3.85	−0.04	3.86	−0.04	0.52%	2.78%
DP6	4.07	−0.03	4.06	−0.03	4.05	−0.03	4.05	−0.03	4.05	−0.03	0.49%	3.85%

Table 8.4.2: Barrier option prices and delta's under Bates

In order to detect which specific setting has a greater contribution to the differences, we have looked at several settings for each product. Though, we do not notice that any specific setting causes a larger risk.

Set	1	2	3	4	5	max. diff.
Nap1	20.46	20.58	20.55	20.56	20.55	0.58%
Nap2	38.15	38.48	38.37	38.45	38.39	0.86%
Nap3	20.49	20.57	20.52	20.58	20.55	0.44%
Nap4	53.31	53.50	53.45	53.51	53.47	0.37%
Nap5	53.98	54.19	54.11	54.16	54.14	0.39%
Nap6	19.53	19.59	19.59	19.57	19.57	0.31%
Nap7	20.85	21.00	21.00	21.04	21.01	0.90%

Table 8.4.3: Napoleon option prices under Bates

8.5 Inter-Model Risk

Inter-model risk is defined as the maximum price difference between all models for a certain derivative. Here we approximate this risk by studying the maximum price difference between the Bates model, the Heston model, the variance gamma model, the Black-Scholes model and the local volatility model. Furthermore, we value the quality of each model for certain properties and try to give a foundation for our choices. For all products inter-model risk is much greater than intra-model risk. In most of the cases the variance gamma model and the Black-Scholes model give very different results. Both the variance gamma model and the Black Scholes model cannot fit the Bates implied volatility surface well, which causes the error.

	Bates	**Heston**	**VG**	**BS**	**LV**	**max. diff.**
Cliquet1	5.57	5.59	8.84	4.51	5.52	48.98%
Cliquet2	12.54	12.62	21.02	8.97	11.61	57.33%
Cliquet3	4.82	4.84	7.76	3.88	4.67	50.00%
Cliquet4	27.83	27.99	27.78	26.59	28.29	6.01%
Cliquet5	2.58	2.61	9.85	1.55	2.86	84.26%
Cliquet6	2.29	2.32	14.28	0.70	3.08	95.10%

Table 8.5.1: Cliquet option prices

If we only consider the remaining three models, then the differences are much smaller. Especially Bates and Heston prices are almost similar. It is interesting to compare the local volatility model to the Bates model as the calibration error is zero in that case. The maximum difference for the cliquets is then 26%, for the discrete knock-out barrier it is 54% and for the Napoleon options it is 39%.

	Bates		**Heston**		**VG**		**BS**		**LV**		**max. diff.**	
	V	Δ	V	Δ	V	Δ	V	Δ	V	Δ	V	Δ
UC1	3.21	0.16	3.20	0.16	9.33	0.69	4.82	0.12	7.01	0.57	65.70%	82.61%
UC2	2.67	0.04	2.64	0.04	2.35	−0.32	2.58	−0.05	2.95	0.29	20.34%	209.40%
UC3	3.34	0.18	3.32	0.17	9.30	0.69	3.93	0.06	6.51	0.55	64.30%	91.30%
UC4	2.03	−0.06	1.99	−0.06	1.85	−0.39	1.44	−0.08	1.75	0.19	29.06%	300.72%
UC5	2.48	0.06	2.48	0.06	4.97	−0.09	4.15	0.04	5.05	0.28	50.89%	131.89%
UC6	2.59	0.06	2.58	0.06	4.07	0.02	3.70	0.04	3.81	0.10	36.61%	80.56%
DP1	4.40	−0.08	4.41	−0.07	0.76	−0.03	2.67	−0.08	2.01	−0.22	82.77%	640.33%
DP2	3.45	0.14	3.40	0.13	1.21	−0.09	2.71	0.01	2.61	−0.31	64.93%	318.29%
DP3	3.34	−0.06	3.33	−0.05	0.62	−0.03	2.06	−0.03	1.58	−0.19	81.44%	530.67%
DP4	2.47	0.13	2.44	0.13	0.97	−0.08	1.61	0.06	1.71	−0.23	60.73%	273.69%
DP5	3.85	−0.04	3.85	−0.04	1.58	−0.03	2.50	−0.06	1.92	−0.08	58.96%	176.67%
DP6	4.05	−0.03	4.05	−0.03	1.90	−0.04	3.02	−0.06	2.46	−0.09	53.09%	210.33%

Table 8.5.2: Barrier option prices and delta's

If we look at the delta's for the barrier options, we notice huge differences. These differences are mostly due to inaccurate approximations in the local volatility model, for which the Monte Carlo method is used. The delta's are not yet converged and the memory of the computer could not handle more simulations.

	Bates	Heston	VG	BS	LV	max. diff.
Nap1	20.55	20.72	27.60	10.76	12.62	61.01%
Nap2	38.37	38.77	52.59	20.52	23.35	60.98%
Nap3	20.52	20.71	31.43	18.26	20.07	41.90%
Nap4	53.45	53.67	60.86	32.08	35.39	47.29%
Nap5	54.11	54.39	61.62	32.51	35.84	47.24%
Nap6	19.59	19.66	24.69	11.60	13.23	53.02%
Nap7	21.00	21.14	25.08	11.97	13.52	52.27%

Table 8.5.3: Napoleon option prices

The Rabo Top Europe Note and the Rabo Deep Autocall Certificate are both not really model dependent. The price differences are much smaller than for the other products and if we compare the Bates model to the local volatility model then the price differences are less than 2% for both products.

	Bates	Heston	VG	BS	LV	max. diff.
RTEN	1.18	1.18	1.15	1.25	1.16	8%
RDAC	1.00	1.00	1.00	1.01	1.00	1%

Table 8.5.4: Prices of the Rabo Top Europe Note (RTEN) and the Rabo Deep Autocall Certificate (RDAC)

Finally, in Table 8.5.5 we present a valuation of certain properties within each model. Here we give marks from 5 for very good to 1 for very poor.

	Bates	Heston	VG	BS	LV
Empirical features	5	4	3	1	2
Implementable	2	3	4	5	5
Fast pricing routines available	3	3	4	5	2
Intuitively understandable	4	4	1	5	3
Calibration	3	4	2	1	5
Hedgeable	2	4	1	4	5

Table 8.5.5: Valuation of model properties

It is well known that jumps are observed in the market, but a more important feature is the stochastic volatility. Bates contains both, so gets the highest mark for the empirical features. Since stochastic volatility is a more important feature than the jumps Heston is the runner up and variance gamma is third. Both Black-Scholes and local volatility do not have stochastic volatility or jumps. Though, local volatility takes into account the current volatility structure and therefore scores higher than the Black-Scholes model.

If we consider the implementability of the models we encounter several problems especially in case of the stochastic volatility models. That is why they have the lowest scores. The mark for the Bates model is even lower, because implementing a PIDE for this model would be very difficult. For Heston only a PDE has to be implemented. Black-Scholes and local volatility are valued best. This is because their implementation is straightforward and easy to understand. Handling the integral in the variance gamma model provides the lower score for this model.

Black-Scholes is the best model if we look at the applicability of fast pricing routines. All methods can be implemented in this framework and simulation can be done analytically, which

saves a lot of time. Variance gamma follows as second. Its only drawback is the integral in the PIDE approach. Heston and Bates are third, because the two-dimensional PDE is much more time consuming, simulations cannot be done analytically and callable options in the quadrature framework become more problematic as well. The local volatility model obtains the lowest grade, because quadrature techniques are not available at all. Furthermore simulations can also not be done analytically.

Black-Scholes is by far the most easy and understandable model. As stochastic volatility and jumps are observed in the market the model can easily be extended to Heston (stochastic volatility) and Heston with Poisson distributed jumps. Variance gamma is very difficult to understand, because in this model stochastic time is introduced to obtain jumps. In this way it is very difficult to give an intuitive meaning to the parameters. The local volatility model looks a lot like the Black-Scholes model. Though, the local volatility function is an artificial function to make sure that all vanilla options can be priced back.

Of course the local volatility model is the best when considering the calibration process. It is easy and the calibration error is zero. Bates is the second model to fit the market data best. This is because it has more free parameters than e.g. Heston. Though, Heston is rated better for this certain property. The reason for this is that the calibration in Bates model is very difficult. Many times an unrealistic solution at the boundaries is obtained. Fourth and fifth are variance gamma and Black-Scholes, respectively. For these models the calibration error becomes quite large.

Hedging can easily be done within the Black-Scholes and local volatility framework. The mark of Black-Scholes is lower, because the hedging parameters are less accurate. Heston has accurate parameters, but an underlying must be hedged with two options. Bates is fourth, because the finite amount of jumps cannot be hedged. Variance gamma is fifth, because hedging an infinite amount of jumps is even more problematic.

Chapter 9

Conclusions and Recommendation for Model Risk

In the third part of this thesis we focused on the determination of model risk. In the definition of model risk, we distinguished intra- and inter-model risk. We defined intra-model risk for a certain contract as the maximum price difference *within one model*, given that the model is adequately calibrated to the initial market prices. These differences arise due to different starting parameters or objective functions in a calibration. Inter model risk is the traditional model risk, i.e. the maximum price difference *over various models*, given that the models are adequately calibrated to the initial market prices.

In Chapter 8.3 we started with the evaluation of intra-model risk. For this purpose we valued several exotic products for different sets of parameters in the Bates model (stochastic volatility with Poisson jumps). These sets of parameters were found by a calibration due to different starting parameters and multiple local minima. The model was calibrated to the EUROSTOXX 50 volatility surface. It turned out that intra-model risk is not a significant contribution to the total risk. Especially when considering realistic sets of parameters, intra-model risk is negligible when comparing it to inter-model risk.

For inter-model risk we compared several exotic option prices under the Bates model, the Heston model, the variance gamma model, the Black-Scholes model and Dupire's local volatility model. All model were calibrated to Bates model, which was itself calibrated to the EUROSTOXX 50 volatility surface. With these calibrated parameters the variance gamma and Black-Scholes model gave completely different option prices. However, both models cannot be calibrated well to the Bates implied volatility surface, which causes a large calibration error. This makes it difficult to determine whether these differences are due to model risk or to calibration risk. Improved calibration algorithms for these models with multiple optimal parameter sets requires further research. This may give better results for e.g. the variance gamma model.

Therefore, in the present financial engineering context it was more interesting to evaluate the remaining three models: Bates, Heston and local volatility. Heston's model could fit the Bates implied volatility surface very well. The calibration error was small and the exotic option prices were almost similar. For the local volatility model the price differences were more significant. The local volatility model is, by definition, calibrated perfectly, i.e. the calibration error is zero. This means that the differences are only due to the model's characteristics. The discrete knock-out barrier option is a model dependent product. Over all considered barriers the maximum price difference between Bates and local volatility was 54%. Other model dependent products are the cliquet and the Napoleon option. Though, price differences are less significant. Over all considered cliquets and Napoleon options the maximum price differences between Bates and

local volatility were 26% and 39%, respectively.

Finally, we conclude that the Heston model is the best model for pricing exotic options. This model contains the most important empirical feature, stochastic volatility. The Bates model also contains this feature and in addition it has jumps, which makes to model even more realistic, but huge drawbacks are that the calibration process is more difficult and the jumps cannot be hedged. In case of Heston a perfect hedge can be formed and the calibration process is easier. The remaining model properties are less important and approximately similar for Bates and Heston. The remaining models gave large calibration errors (variance gamma and Black-Scholes) or did not have the important empirical features (Black-Scholes and local volatility).

We end this chapter with some suggestions for further research:

- In this thesis we used a heuristic calibration technique. We suggest to improve the calibration process by using more advanced optimization techniques, e.g. genetic algorithms or Krieging. We also suggest to get a grip on the optimization techniques used in standard programs like Microsoft Excel. These techniques seem to find a local minima, but we do not have a clue on how these methods function.

- Many different payoffs exist and probably many of those can be calculated with quadrature techniques. It would be interesting the study the applicability of quadrature techniques to more kinds of exotic payoffs. And it would be interesting to examine the model risk for those payoffs.

- We noticed that the variance gamma model could not be calibrated well to the Bates implied volatility surface. This may be due to the dynamics, but it has also less free parameters than the Bates model. Another purely jump model is the CGMY model, which has more free parameters than the variance gamma model. We suggest to consider more different models, e.g. the CGMY model, to obtain a better view on which features are important in a model.

Bibliography

Abramowitz, M. and Stegun, I. (1972). *Handbook of Mathematical Functions*, Dover Publications, New York, USA.

Almendral, A. and Oosterlee, C. W. (2005). Numerical valuation of options with jumps in the underlying, *Applied Numerical Mathematics* **53**: 1–18.

Almendral, A. and Oosterlee, C. W. (2006). On American options under the variance gamma process, *Working paper*, to appear in Applied Mathematical Finance.

Andricopoulos, A., Widdicks, M., Duck, P. and Newton, D. (2003). Universal option valuation using quadrature methods, *Journal of Financial Economics* **67**: 447–471.

Andricopoulos, A., Widdicks, M., Duck, P. and Newton, D. (2004). Extending quadrature methods to value multi-asset and complex path-dependent options, *Working paper*, University of Manchester.

Bakshi, G. and Madan, D. (2000). Spanning and derivative security valuation, *Journal of Financial Economics* **55**: 205–238.

Bates, D. (1996). Jump and stochastic volatility: Exchange rate processes implicit in deutsche mark options, *Review of Financial Studies* **9**: 69–107.

Black, F. and Scholes, M. (1973). The pricing of options and corporate liabilities, *Journal of Political Economy* **81**: 637–659.

Boyle, P. (1977). Options: A Monte Carlo approach, *Journal of Financial Economics* **4**: 323–338.

Broadie, M. and Kaya, O. (2004). Exact simulation of stochastic volatility and other affine jump diffusion processes, *Working paper*, Columbia University.

Buitelaar, J. (2006). Control variates for callable libor exotics, *Msc. thesis*, Delft University of Technology.

Burden, R. L. and Faires, J. D. (2001). *Numerical Analysis*, 7th edn, Brooks/Cole, USA.

Carr, P. and Madan, D. (1999). Option valuation using the fast Fourier transform, *Journal of Computational Finance* **3**: 463–520.

Chang, C.-C., Chung, S.-L. and Stapleton, R. C. (2002). Richardson extrapolation technique for pricing American-style options, *EFMA 2002 London Meetings* .

Cont, R. and Tankov, P. (2004). *Financial Modeling With Jump Processes*, 1st edn, Chapman & Hall/CRC.

Cox, J., Ingersoll, J. and Ross, S. (1985). A theory of the term structure of interest rates, *Econometrica* **53**: 385–408.

Cox, J. and Ross, S. (1976). The valuation of options for alternitive stochastic processes, *Journal of Financial Economics* **3**: 145–166.

Den Iseger, P. and Oldenkamp, E. (2005). Cliquet options: Pricing and Greeks in deterministic and stochastic volatility models, *Working paper*, Submitted to Finance and Stochastics.

Derman, E., Kani, I., Ergener, D. and Bardhan, I. (1995). Enhanced Numerical Methods for Options with Barriers, *Quantitative Strategies Research Notes*, Goldman Sachs.

Detlefsen, K. and Härdle, W. K. (2006). Calibration risk for exotic options, *Working paper*, Center for Applied Statistics and Economics.

Duffie, D., Pan, J. and Singleton, K. (2000). Transform analysis and asset pricing for affine jump-diffusions, *Econometrica* **68**: 1343–1376.

Dupire, B. (1994). Pricing with a smile, *Risk Magazine* **7**: 18–20.

Feller, W. (1951). Two singular diffusion problems, *Annals of Mathematics* **54**: 173–182.

Geske, R. and Johnson, H. (1984). The American put valued analytically, *Journal of Finance* **39**: 1511–1542.

Heston, S. (1993). A closed-form solution for options with stochastic volatility with applications to bond and currency options, *Review of Financial Studies* **6**: 327–343.

Hull, J. and Suo, W. (2002). A methodology for assessing model risk and its application to the implied volatility model, *Journal of Financial and Quatitative Analysis* **37**(2): 297–318.

Ito, K. and Toivanen, J. (2006). Lagrange multiplier approach with optimized finite difference stencils for pricing American options under stochastic volatility, *Scientific report*, Department of Mathematical Information Technology, University of Jyväskylä, Finland.

Kloeden, P. E. and Platen, E. (1992). *Numerical Solution of Stochastic Differential Equations*, 1st edn, Springer Verlag, Berlin.

Longstaff, F. and Schwartz, E. (2001). Valuing American options by simulation: A simple least squares approach, *Review of Financial Studies* **14**(1): 113–147.

Lord, R. (2005). *private communication.*

Lord, R. and Kahl, C. (2006). Why the rotation count algorithm works, *Working paper.*

Lord, R., Koekkoek, R. and van Dijk, D. (2006). A comparison of biased simulation schemes for stochastic volatility models, *Working paper.*

Madan, D. (2005). Financial innovation and new structured products in the equity world, *workshop given on 30 november 2005*, Carisma and Brunel University.

Madan, D. B. and Seneta, E. (1990). The variance gamma (v.g.) model for share market returns, *Journal of Business* **63**: 511–524.

Ng, M. (2005). Option pricing via the FFT, *Msc. thesis*, Delft University of Technology.

O'Sullivan, C. (2004). Path dependent option pricing under Lévy processes: Applied to Bermudan options, *Working paper*, University College Dublin.

Press, W. H., Teukolsky, S. A., Vetterling, W. T. and Flannery, B. P. (1992). *Numerical Recipes in C: The Art of Scientific Computing*, 2nd edn, Cambridge University Press, New York, USA.

Wilmott, P. (1998). *Derivatives: The Theory and Practice of Financial Engineering*, John Wiley & Sons Ltd, Chichester, UK.

Appendix A

Derivation of the Local Volatility Function

In this section we derive the local volatility function. Let us start with the representation of two important results in finance, which are necessary for the derivation.

Theorem 1 *(Feynman-Kac) Assume that F is a solution to the boundary value problem*

$$\begin{aligned} \frac{\partial F}{\partial t}+\tfrac{1}{2}\sigma\left(x,t\right)^{2}\frac{\partial^{2}F}{\partial x^{2}}+\mu\left(x,t\right)\frac{\partial F}{\partial x}-rF &= 0, && \text{(A.0.1)}\\ F\left(x,T\right) &= g\left(x_{T}\right). && \text{(A.0.2)} \end{aligned}$$

Assume furthermore that the process $\sigma\left(x\left(t\right),t\right)\frac{\partial F}{\partial x}$ is in $\mathcal{L}^{2}$, with x defined below, then F has the representation as follows:

$$F\left(x,t\right)=e^{-r(T-t)}\mathbb{E}_{t}\left[g\left(x_{T}\right)\right], \tag{A.0.3}$$

where x satisfies the SDE

$$\begin{aligned} dx\left(s\right) &= \mu\left(x\left(s\right),s\right)ds+\sigma\left(x\left(s\right),s\right)dW\left(s\right), && \text{(A.0.4)}\\ x\left(t\right) &= x. && \text{(A.0.5)} \end{aligned}$$

Proof. For any twice differentiable function ϕ Itô's lemma gives

$$d\left(\phi\left(x\left(s\right),s\right)\right)=\left\{\frac{\partial\phi}{\partial s}+\mu\left(x\left(s\right),s\right)\frac{\partial\phi}{\partial x}+\frac{1}{2}\sigma\left(x\left(s\right),s\right)^{2}\frac{\partial^{2}\phi}{\partial x^{2}}\right\}ds+\sigma\left(x\left(s\right),s\right)\frac{\partial\phi}{\partial x}dW\left(s\right).$$

When choosing $\phi=e^{r(T-s)}F\left(x\left(s\right),s\right)$, with $F\left(x\left(s\right),s\right)$ the solution of (A.0.1), we get

$$d\left(e^{r(T-s)}F\left(x\left(s\right),s\right)\right)=\sigma\left(x\left(s\right),s\right)e^{r(T-s)}\frac{\partial F}{\partial x}dW\left(s\right).$$

Integrating and taking the expected value proves the theorem:

$$\begin{aligned} \mathbb{E}_{t}\left[\int_{t}^{T}d\left(e^{r(T-s)}F\left(x\left(s\right),s\right)\right)\right] &= \mathbb{E}_{t}\left[\int_{t}^{T}\sigma\left(x\left(s\right),s\right)e^{r(T-s)}\frac{\partial F}{\partial x}dW\left(s\right)\right],\\ \mathbb{E}_{t}\left[F\left(x\left(T\right),T\right)-e^{r(T-t)}F\left(x\left(t\right),t\right)\right] &= 0,\\ \mathbb{E}_{t}\left[F\left(x\left(T\right),T\right)\right]-e^{r(T-t)}\mathbb{E}_{t}\left[F\left(x\left(t\right),t\right)\right] &= 0,\\ \mathbb{E}_{t}\left[F\left(x\left(t\right),t\right)\right] &= e^{-r(T-t)}\mathbb{E}_{t}\left[g\left(x_{T}\right)\right],\\ F\left(x\left(t\right),t\right) &= e^{-r(T-t)}\mathbb{E}_{t}\left[g\left(x_{T}\right)\right]. \end{aligned}$$

■

Theorem 2 *(Kolmogorov forward equation) Assume that x is a solution to the equation*

$$dx(t) = \mu(x(t), t)\, dt + \sigma(x(t), t)\, dW(t), \tag{A.0.6}$$

where $\mu(x(t), t)$ and $\sigma(x(t), t)$ are bounded functions. Assume further that x has a transition density $p(x(s), s; x(t), t)$. Then p satisfies the Kolmogorov forward equation

$$\frac{\partial p}{\partial t} = \tfrac{1}{2}\frac{\partial^2}{\partial x^2}\left[\sigma(x(t), t)^2 p\right] - \frac{\partial}{\partial x}\left[\mu(x(t), t) p\right], \ (x, t) \in \mathbb{R} \times (0, T), \tag{A.0.7}$$

$$p(x(s), s; x(t), t) \to \delta_{x(s)}, \quad \text{as } t \downarrow s, \tag{A.0.8}$$

where δ_y is the Dirac delta function.

Proof. Consider a twice differentiable function $h(y)$, with the condition that $h(y) = h'(y) = 0$ for $y \leq 0$ and $\lim_{y\to\infty} h(y) = \lim_{y\to\infty} h'(y) = 0$. Itô's lemma holds for any twice differentiable function. Therefore we have

$$dh(x(t)) = \left\{\mu(x(t), t)\frac{\partial h}{\partial x} + \tfrac{1}{2}\sigma(x(t), t)^2 \frac{\partial^2 h}{\partial x^2}\right\} dt + \sigma(x(t), t)\frac{\partial h}{\partial x} dW(t).$$

So,

$$h(x(t)) = h(x(s)) + \int_s^t \left\{\mu(x(u), u)\frac{\partial h}{\partial x} + \tfrac{1}{2}\sigma(x(u), u)^2 \frac{\partial^2 h}{\partial x^2}\right\} du + \int_s^t \sigma(x(u), u)\frac{\partial h}{\partial x} dW(u),$$

and

$$\mathbb{E}_s[h(x(t))] = h(x(s)) + \mathbb{E}_s\left[\int_s^t \left\{\mu(x(u), u)\frac{\partial h}{\partial x} + \tfrac{1}{2}\sigma(x(u), u)^2 \frac{\partial^2 h}{\partial x^2}\right\} du\right],$$

or equivalently,

$$\int_0^\infty h(x(t))\, p dx(t) = h(x(s)) + \int_s^t \int_0^\infty \mu(x(u), u)\frac{\partial h}{\partial x} p dx(u)\, du + \tfrac{1}{2}\int_s^t \int_0^\infty \sigma(x(u), u)^2 \frac{\partial^2 h}{\partial x^2} p dx(u)\, du.$$

Differentiate with respect to t to get

$$\int_0^\infty h(x(t))\frac{\partial p}{\partial t} dx(t) = \int_0^\infty \mu(x(t), t)\frac{\partial h}{\partial x} p dx(t) + \tfrac{1}{2}\int_0^\infty \sigma(x(t), t)^2 \frac{\partial^2 h}{\partial x^2} p dx(t).$$

Integrating by parts yields

$$\int_0^\infty \mu(x(t), t)\frac{\partial h}{\partial x} p dx(t) = \underbrace{\left[h(x(t))\mu(x(t), t) p\right]_{x(t)=0}^\infty}_{=0} - \int_0^\infty h(x(t))\frac{\partial}{\partial x}\left[\mu(x(t), t) p\right] dx(t)$$

and

$$\begin{aligned}
\int_0^\infty \sigma(x(t), t)^2 \frac{\partial^2 h}{\partial x^2} p dx(t) &= \underbrace{\left[\frac{\partial h}{\partial x}\sigma(x(t), t)^2 p\right]_{x(t)=0}^\infty}_{=0} - \int_0^\infty \frac{\partial h}{\partial x}\frac{\partial}{\partial x}\left[\sigma(x(t), t)^2 p\right] dx(t), \\
&= \underbrace{\left[h(x(t))\frac{\partial}{\partial x}\left[\sigma(x(t), t)^2 p\right]\right]_{x(t)=0}^\infty}_{=0} \\
&\qquad + \int_0^\infty h(x(t))\frac{\partial^2}{\partial x^2}\left[\sigma(x(t), t)^2 p\right] dx(t).
\end{aligned}$$

Therefore,

$$\int_0^\infty h(x(t))\frac{\partial p}{\partial t}dx(t) = -\int_0^\infty h(x(t))\frac{\partial}{\partial x}\left[\mu(x(t),t)\,p\right]dx(t) + \tfrac{1}{2}\int_0^\infty h(x(t))\frac{\partial^2}{\partial x^2}\left[\sigma(x(t),t)^2 p\right]dx(t),$$

or equivalently,

$$\int_0^\infty h(x(t))\left[\frac{\partial p}{\partial t} + \frac{\partial}{\partial x}\left[\mu(x(t),t)\,p\right] - \tfrac{1}{2}\frac{\partial^2}{\partial x^2}\left[\sigma(x(t),t)^2 p\right]\right]dx(t) = 0.$$

The last equation holds for every function h that satisfies the conditions described at the beginning of the proof. It implies that

$$\frac{\partial p}{\partial t} + \frac{\partial}{\partial x}\left[\mu(x(t),t)\,p\right] - \tfrac{1}{2}\frac{\partial^2}{\partial x^2}\left[\sigma(x(t),t)^2 p\right] = 0.$$

■

Recall that the PDE in the local volatility model reads, see (1.5.3),

$$\frac{\partial V}{\partial \tau} = \tfrac{1}{2}\sigma(x,\tau)^2\frac{\partial^2 V}{\partial x^2} + \left(r - \tfrac{1}{2}\sigma(x,\tau)^2\right)\frac{\partial V}{\partial x} - rV. \tag{A.0.9}$$

where $\tau = T - t$. Denote $u(x,t) = V(x,\tau)$, then we have

$$\frac{\partial u}{\partial t} + \tfrac{1}{2}\sigma(x,t)^2\frac{\partial^2 u}{\partial x^2} + \left(r - \tfrac{1}{2}\sigma(x,t)^2\right)\frac{\partial u}{\partial x} - ru = 0, \tag{A.0.10}$$

with the final condition $u(x,T) = g(x_T)$. According to the Feynman-Kac theorem the option price $u(x,t)$ can be calculated as

$$u(x,t) = e^{-r(T-t)}\int_{-\infty}^{\infty} g(x_T)\,p(x,t;x_T,T)\,dx_T. \tag{A.0.11}$$

In particular, consider the present value of a call option. We have that

$$C(x,k,t) = e^{-r(T-t)}\int_k^\infty \left(e^{x_T} - e^k\right)p(x,t;x_T,T)\,dx_T, \tag{A.0.12}$$

where $k = \log K$. We would like to differentiate equation (A.0.12). For this we use the following lemma.

Lemma 3 *Consider a function $f(x,k)$, which is continuous in x and differentiable in k. Furthermore, assume that $\int_k^\infty f(x,k)\,dx < \infty$. Then we have:*

$$\frac{\partial}{\partial k}\int_k^\infty f(x,k)\,dx = \int_k^\infty \frac{\partial f(x,k)}{\partial k}dx - f(k,k). \tag{A.0.13}$$

Proof.

$$\begin{aligned}
\frac{\partial}{\partial k}\int_{k}^{\infty} f(x,k)\,dx &= \lim_{\Delta\to 0}\frac{\int_{k+\Delta}^{\infty} f(x,k+\Delta)\,dx - \int_{k}^{\infty} f(x,k)\,dx}{\Delta} \\
&= \lim_{\Delta\to 0}\frac{\int_{k+\Delta}^{\infty} f(x,k+\Delta) - f(x,k)\,dx - \int_{k}^{k+\Delta} f(x,k)\,dx}{\Delta} \\
&= \lim_{\Delta\to 0}\frac{\int_{k+\Delta}^{\infty} f(x,k+\Delta) - f(x,k)\,dx}{\Delta} - \lim_{\Delta\to 0}\frac{\int_{k}^{k+\Delta} f(x,k)\,dx}{\Delta}.
\end{aligned}$$

Since $f(x,k)$ is continuous in x there exists a function $F(x,k)$ such that: $\frac{\partial F}{\partial x}(x,k) = f(x,k)$. Therefore,

$$\begin{aligned}
\frac{\partial}{\partial k}\int_{k}^{\infty} f(x,k)\,dx &= \lim_{\Delta\to 0}\frac{\lim_{y\to\infty} F(y,k+\Delta) - F(k+\Delta,k+\Delta) - F(y,k) + F(k+\Delta,k)}{\Delta} \\
&\qquad - \lim_{\Delta\to 0}\frac{F(k+\Delta,k) - F(k,k)}{\Delta} \\
&= \lim_{y\to\infty}\frac{\partial F}{\partial k}(y,k) - \frac{\partial F}{\partial k}(k,k) - \frac{\partial F}{\partial x}(k,k),
\end{aligned}$$

where $\frac{\partial F}{\partial k}(k,k)$ is the derivative to the second argument and $\frac{\partial F}{\partial x}(k,k)$ the derivative to the first argument.

$$\begin{aligned}
\frac{\partial}{\partial k}\int_{k}^{\infty} f(x,k)\,dx &= \int_{k}^{\infty}\frac{\partial F(x,k)}{\partial k\partial x}dx - f(k,k) \\
&= \int_{k}^{\infty}\frac{\partial f(x,k)}{\partial k}dx - f(k,k).
\end{aligned}$$

■

With this result we can easily calculate the first and second derivatives to k of (A.0.12):

$$\frac{\partial}{\partial k}C(x,k,t) = -e^{-r(T-t)}\int_{k}^{\infty} e^{k} p(x,t;x_T,T)\,dx_T, \tag{A.0.14}$$

$$\begin{aligned}
\frac{\partial^2}{\partial k^2}C(x,k,t) &= -e^{-r(T-t)}\int_{k}^{\infty} e^{k} p(x,t;x_T,T)\,dx_T + e^{-r(T-t)}e^{k} p(x,t;k,T) \\
&= \frac{\partial}{\partial k}C(x,k,t) + e^{-r(T-t)}e^{k} p(x,t;k,T)
\end{aligned} \tag{A.0.15}$$

Now, from Theorem 2 we know that this transition density must satisfy the Kolmogorov forward equation. Thus $p(x(s),s;x(t),t)$ solves

$$\tfrac{1}{2}\frac{\partial^2}{\partial x_T^2}\left[\sigma(x_T,T)^2 p\right] - \frac{\partial}{\partial x_T}\left[\left(\mu - \tfrac{1}{2}\sigma(x(t),t)^2\right)p\right] - \frac{\partial p}{\partial T} = 0. \tag{A.0.16}$$

We multiply this by $e^{-r(T-t)}\left(e^{x_T}-e^k\right)^+$ and integrate the comple expression, giving

$$\begin{aligned}
&\tfrac{1}{2}e^{-r(T-t)}\int_k^\infty \frac{\partial^2}{\partial x_T^2}\left[\sigma\left(x_T,T\right)^2 p\right]\left(e^{x_T}-e^k\right)dx_T \\
&-e^{-r(T-t)}\int_k^\infty \frac{\partial}{\partial x_T}\left[\left(\mu-\tfrac{1}{2}\sigma\left(x\left(t\right),t\right)^2\right)p\right]\left(e^{x_T}-e^k\right)dx_T - e^{-r(T-t)}\int_k^\infty \frac{\partial p}{\partial T}\left(e^{x_T}-e^k\right)dx_T = 0.
\end{aligned} \tag{A.0.17}$$

The first term can be integrated by parts to give

$$\begin{aligned}
&\tfrac{1}{2}\int_k^\infty \frac{\partial^2}{\partial x_T^2}\left[\sigma\left(x_T,T\right)^2 p\right]\left(e^{x_T}-e^k\right)dx_T \\
&= \tfrac{1}{2}\left\{\underbrace{\left[\frac{\partial}{\partial x_T}\left[\sigma\left(x_T,T\right)^2 p\right]\left(e^{x_T}-e^k\right)\right]_{x_T=k}^\infty}_{=0} - \int_k^\infty \frac{\partial}{\partial x_T}\left[\sigma\left(x_T,T\right)^2 p\right]e^{x_T}dx_T\right\} \\
&= \tfrac{1}{2}\left\{-\left[\sigma\left(x_T,T\right)^2 pe^{x_T}\right]_{x_T=k}^\infty + \int_k^\infty \sigma\left(x_T,T\right)^2 pe^{x_T}dx_T\right\} \\
&= \tfrac{1}{2}\sigma\left(k,T\right)^2 pe^k + \tfrac{1}{2}\int_k^\infty \sigma\left(x_T,T\right)^2 pe^{x_T}dx_T.
\end{aligned} \tag{A.0.18}$$

The second term in (A.0.17) can be integrated by parts to give

$$\begin{aligned}
&-\int_k^\infty \frac{\partial}{\partial x}\left[\left(\mu-\tfrac{1}{2}\sigma\left(x\left(T\right),T\right)^2\right)p\right]\left(e^{x_T}-e^k\right)dx_T \\
&= -\underbrace{\left[\left[\left(\mu-\tfrac{1}{2}\sigma\left(x\left(T\right),T\right)^2\right)p\right]\left(e^{x_T}-e^k\right)\right]_{x_T=k}^\infty}_{=0} + \int_k^\infty \left[\left(\mu-\tfrac{1}{2}\sigma\left(x\left(T\right),T\right)^2\right)p\right]e^{x_T}dx_T \\
&= \int_k^\infty \mu pe^{x_T}dx_T - \int_k^\infty \tfrac{1}{2}\sigma\left(x\left(T\right),T\right)^2 pe^{x_T}dx_T.
\end{aligned} \tag{A.0.19}$$

The last term in equation (A.0.17) can be integrated directly giving

$$\begin{aligned}
-e^{-r(T-t)}\int_k^\infty \frac{\partial p}{\partial T}\left(e^{x_T}-e^k\right)dx_T &= -e^{-r(T-t)}\frac{\partial}{\partial T}\int_k^\infty p\left(e^{x_T}-e^k\right)dx_T \\
&= -e^{-r(T-t)}\frac{\partial}{\partial T}\left[e^{r(T-t)}C\left(x,k,t\right)\right] = -\frac{\partial}{\partial T}C\left(x,k,t\right) - rC\left(x,k,t\right).
\end{aligned} \tag{A.0.20}$$

So, when substituting (A.0.18), (A.0.19) and (A.0.20) into (A.0.17) we obtain

$$\begin{aligned}
&\tfrac{1}{2}e^{-r(T-t)}\sigma\left(k,T\right)^2 pe^k + \tfrac{1}{2}e^{-r(T-t)}\int_k^\infty \sigma\left(x_T,T\right)^2 pe^{x_T}dx_T \\
&+e^{-r(T-t)}\int_k^\infty \mu pe^{x_T}dx_T - \tfrac{1}{2}e^{-r(T-t)}\int_k^\infty \sigma\left(x\left(T\right),T\right)^2 pe^{x_T}dx_T - \frac{\partial}{\partial T}C\left(x,k,t\right) - rC\left(x,k,t\right) = 0,
\end{aligned}$$

$$\tfrac{1}{2}e^{-r(T-t)}\sigma\left(k,T\right)^{2}pe^{k}+e^{-r(T-t)}\int_{k}^{\infty}\mu pe^{x_{T}}dx_{T}-\frac{\partial}{\partial T}C\left(x,k,t\right)-rC\left(x,k,t\right)=0.$$

Equations (A.0.12), (A.0.14) and (A.0.15) can be used to make a substitution giving

$$\tfrac{1}{2}\sigma\left(k,T\right)^{2}\left(\frac{\partial^{2}}{\partial k^{2}}C\left(x,k,t\right)-\frac{\partial}{\partial k}C\left(x,k,t\right)\right)$$
$$+\mu\left(C\left(x,k,t\right)-\frac{\partial}{\partial k}C\left(x,k,t\right)\right)-\frac{\partial}{\partial T}C\left(x,k,t\right)-rC\left(x,k,t\right)=0.$$

Re-arranging this gives the local volatility function in terms of k and T:

$$\sigma\left(k,T\right)^{2}=2\frac{\frac{\partial}{\partial T}C\left(x,k,t\right)+\mu\frac{\partial}{\partial k}C\left(x,k,t\right)+\left(r-\mu\right)C\left(x,k,t\right)}{\frac{\partial^{2}}{\partial k^{2}}C\left(x,k,t\right)-\frac{\partial}{\partial k}C\left(x,k,t\right)}. \tag{A.0.21}$$

After substitution of x for k and t for T, we find the appropriate local volatility function:

$$\sigma\left(x,t\right)^{2}=2\frac{\frac{\partial}{\partial T}C\left(x,x,t\right)+\mu\frac{\partial}{\partial k}C\left(x,x,t\right)+\left(r-\mu\right)C\left(x,x,t\right)}{\frac{\partial^{2}}{\partial k^{2}}C\left(x,x,t\right)-\frac{\partial}{\partial k}C\left(x,x,t\right)}. \tag{A.0.22}$$

Appendix B

Derivation of the CCF of Heston

The derivation of the characteristic function of the logarithm of the stock given the initial variance and given the variance at maturity is presented is this section. For this derivation we write the dynamics of the Heston model in a different manner:

$$dx(t) = \left(r - \frac{1}{2}v(t)\right)dt + \rho\sqrt{v(t)}dW_1(t) + \sqrt{1-\rho^2}\sqrt{v(t)}dW_2(t), \quad \text{(B.0.1)}$$

$$dv(t) = -\lambda(v(t) - \overline{v})\,dt + \eta\sqrt{v(t)}dW_1(t). \quad \text{(B.0.2)}$$

From (B.0.1) and (B.0.2) it follows that

$$x_T = x_0 + r(T-t_0) - \frac{1}{2}\int_{t_0}^{T} v(s)\,ds + \rho\int_{t_0}^{T}\sqrt{v(s)}dW_1(s) + \sqrt{1-\rho^2}\int_{t_0}^{T}\sqrt{v(s)}dW_2(s), \quad \text{(B.0.3)}$$

$$v_T = v_0 + \lambda\overline{v}(T-t_0) - \lambda\int_{t_0}^{T} v(s)\,ds + \eta\int_{t_0}^{T}\sqrt{v(s)}dW_1(s). \quad \text{(B.0.4)}$$

Rewrite (B.0.4) as

$$\eta\int_{t_0}^{T}\sqrt{v(s)}dW_1(s) = v_T - v_0 - \lambda\overline{v}(T-t_0) + \lambda\int_{t_0}^{T} v(s)\,ds \quad \text{(B.0.5)}$$

and substitution of (B.0.5) into (B.0.3) gives

$$\begin{aligned} x_T - x_0 &= r(T-t_0) - \frac{1}{2}\int_{t_0}^{T} v(s)\,ds \\ &\quad + \frac{\rho}{\eta}\left(v_T - v_0 - \lambda\overline{v}(T-t_0) + \lambda\int_{t_0}^{T} v(s)\,ds\right) + \sqrt{1-\rho^2}\int_{t_0}^{T}\sqrt{v(s)}dW_2(s) \\ &= r(T-t_0) + \frac{\rho}{\eta}\left(v_T - v_0 - \lambda\overline{v}(T-t_0)\right) + \left(\frac{\lambda\rho}{\eta} - \frac{1}{2}\right)\int_{t_0}^{T} v(s)\,ds \\ &\quad + \sqrt{1-\rho^2}\int_{t_0}^{T}\sqrt{v(s)}dW_2(s). \end{aligned}$$

Notice that $\int_{t_0}^{T}\sqrt{v(s)}dW_2(s)$ is normally distributed with zero expectation and variance equal to $\int_{t_0}^{T} v(s)\,ds$. Denote by Z the standard normal cumulative distribution function, then the

characteristic function $\widehat{p}(\omega)$ conditional on v_0 and v_T reads

$$\begin{aligned}
\mathbb{E}_{t_0}\left[e^{i\omega(x_T-x_0)}\middle| v_T\right] &= \mathbb{E}_{t_0}\left[\exp\left(i\omega\left[r(T-t_0)+\frac{\rho}{\eta}(v_T-v_0-\lambda\overline{v}(T-t_0))\right.\right.\right.\\
&\qquad\left.\left.\left.+\left(\frac{\lambda\rho}{\eta}-\frac{1}{2}\right)\int_{t_0}^{T}v(s)\,ds+\sqrt{1-\rho^2}\sqrt{\int_{t_0}^{T}v(s)\,ds}Z\right]\right)\middle| v_T\right]\\
&= \exp\left(i\omega\left[r(T-t_0)+\frac{\rho}{\eta}(v_T-v_0-\lambda\overline{v}(T-t_0))\right]\right)\\
&\qquad\times\mathbb{E}_{t_0}\left[\exp\left(i\omega\left(\frac{\lambda\rho}{\eta}-\frac{1}{2}\right)\int_{t_0}^{T}v(s)\,ds+i\omega\sqrt{1-\rho^2}\sqrt{\int_{t_0}^{T}v(s)\,ds}Z\right)\middle| v_T\right]\\
&= \exp\left(i\omega\left[r(T-t_0)+\frac{\rho}{\eta}(v_T-v_0-\lambda\overline{v}(T-t_0))\right]\right)\\
&\qquad\times\mathbb{E}_{t_0}\left[\exp\left(i\omega\left(\frac{\lambda\rho}{\eta}-\frac{1}{2}\right)V+i\omega\sqrt{1-\rho^2}\sqrt{V}Z\right)\middle| v_T\right],
\end{aligned}$$

where $V:=\int_{t_0}^{T}v(s)\,ds$. Using the tower property for expectations we obtain

$$\begin{aligned}
\mathbb{E}_{t_0}\left[e^{i\omega(x_T-x_0)}\middle| v_T\right] &= \exp\left(i\omega\left[r(T-t_0)+\frac{\rho}{\eta}(v_T-v_0-\lambda\overline{v}(T-t_0))\right]\right)\\
&\qquad\times\mathbb{E}_{t_0}\left[\mathbb{E}_{t_0}\left[\exp\left(i\omega\left(\frac{\lambda\rho}{\eta}-\frac{1}{2}\right)V+i\omega\sqrt{1-\rho^2}\sqrt{V}Z\right)\middle| v_T,V\right]\right]\\
&= \exp\left(i\omega\left[r(T-t_0)+\frac{\rho}{\eta}(v_T-v_0-\lambda\overline{v}(T-t_0))\right]\right)\\
&\qquad\times\mathbb{E}_{t_0}\left[\exp\left(i\omega\left(\frac{\lambda\rho}{\eta}-\frac{1}{2}\right)V-\frac{1}{2}\omega^2\left(1-\rho^2\right)V\right)\middle| v_T\right]\\
&= \exp\left(i\omega\left[r(T-t_0)+\frac{\rho}{\eta}(v_T-v_0-\lambda\overline{v}(T-t_0))\right]\right)\\
&\qquad\times\mathbb{E}_{t_0}\left[\exp\left(i\left[\omega\left(\frac{\lambda\rho}{\eta}-\frac{1}{2}\right)+\frac{1}{2}i\omega^2\left(1-\rho^2\right)\right]\int_{t_0}^{T}v(s)\,ds\right)\middle| v_T\right]\\
&= \exp\left(i\omega\left[r(T-t_0)+\frac{\rho}{\eta}(v_T-v_0-\lambda\overline{v}(T-t_0))\right]\right) \qquad \text{(B.0.6)}\\
&\qquad\times\Phi\left(\omega\left(\frac{\lambda\rho}{\eta}-\frac{1}{2}\right)+\frac{1}{2}i\omega^2\left(1-\rho^2\right)\right),
\end{aligned}$$

with $\Phi(a)$ the characteristic function of $\int_{t_0}^{T} v(s)\,ds$ given v_0 and given v_T. In Broadie and Kaya (2004) it has been shown that

$$\begin{aligned}\Phi(a) &= \frac{\gamma(a)\,e^{-\frac{1}{2}(\gamma(a)-\lambda)(T-t_0)}\left(1-e^{-\lambda(T-t_0)}\right)}{\lambda\left(1-e^{-\gamma(a)(T-t_0)}\right)} \\ &\quad\times\exp\left(\frac{v_0+v_T}{\eta^2}\left[\frac{\lambda\left(1+e^{-\lambda(T-t_0)}\right)}{1-e^{-\lambda(T-t_0)}}-\frac{\gamma(a)\left(1+e^{-\gamma(a)(T-t_0)}\right)}{1-e^{-\theta(a)(T-t_0)}}\right]\right) \\ &\quad\times\frac{I_{\frac{1}{2}d-1}\left[\sqrt{v_0v_T}\frac{4\gamma(a)e^{-\frac{1}{2}\gamma(a)(T-t_0)}}{\eta^2\left(1-e^{-\gamma(a)(T-t_0)}\right)}\right]}{I_{\frac{1}{2}d-1}\left[\sqrt{v_0v_T}\frac{4\lambda e^{-\frac{1}{2}\lambda(T-t_0)}}{\eta^2\left(1-e^{-\lambda(T-t_0)}\right)}\right]},\end{aligned} \tag{B.0.7}$$

where $\gamma(a)=\sqrt{\lambda^2-2\eta^2ia}$, $d=4\overline{v}\lambda/\eta^2$ and $I_\nu(x)$ is the modified Bessel function of the first kind.

Appendix C

Rabo Top Europe Note

Underlying index	Dow Jones EUROSTOXX 50 Index
Fixing Date Launch Date Issue/ Payment Date	9 June 2006 12 June 2006 16 June 2006
Final Valuation Date	9 June 2014
Maturity Date	16 June 2014
Notional	EUR 100
Barrier Level	60% of the closing value of the Underlying Index on the Fixing Date
Redemption per Note in Cash	On the Maturity Date there are two possible redemption scenarios: 1) If $Index_{final} \geq H$, the Notes will be redeemed according to the following formula: $N\frac{Index_{\max}}{Index_{init}}$ 2) If $Index_{final} < H$, the Notes will be redeemed according to the following formula: $N\frac{Index_{final}}{Index_{init}}$ where: - $Index_{\max}$: The highest recorded closing value of the Underlying Index on any of the Monthly Observation Dates - $Index_{init}$: The closing value of the Underlying Index on the Fixing Date - $Index_{final}$: The closing value of the Underlying Index on the Final Valuation Date - N: The Notional Amount - H: The Barrier Level
Monthly Observation Dates	9th of each Month, with the first on the Fixing Date and the last on the Final Valuation Date: 97 Observations If a Valuation Date is not a Business Day, the next Business Day will be the Valuation Date

Table C.0.1: Rabo Top Europe Note

Appendix D

Rabo Deep Autocall Certificate

Underlying index	Dow Jones EUROSTOXX 50 Index
Fixing Date/ Payment Date	6 June 2006/ 13 June 2006
Final Valuation Date	9 July 2012
Maturity Date	11 July 2012
Notional	EUR 100
Early Redemption in % per certificate	Certificates will be redeemed early: At 108.45% on $DATE1$ if $INDEX1 \geq 90\%$ of the $INDEX_{init}$ At 116.90% on $DATE2$ if $INDEX2 \geq 90\%$ of the $INDEX_{init}$ At 125.35% on $DATE3$ if $INDEX3 \geq 90\%$ of the $INDEX_{init}$ At 133.80% on $DATE4$ if $INDEX4 \geq 90\%$ of the $INDEX_{init}$ At 142.25% on $DATE5$ if $INDEX5 \geq 90\%$ of the $INDEX_{init}$ where: - $DATE1$: 20 June 2007 - $DATE2$: 18 June 2008 - $DATE3$: 24 June 2009 - $DATE4$: 30 June 2010 - $DATE5$: 6 July 2011 - $INDEX1$: The closing price of the index on 18 June 2007 - $INDEX2$: The closing price of the index on 16 June 2008 - $INDEX3$: The closing price of the index on 22 June 2009 - $INDEX4$: The closing price of the index on 28 June 2010 - $INDEX5$: The closing price of the index on 4 July 2011
Final Redemption in % per certificate	If the certificate does not get called early, the Final Redemption will be as follows: 1) If $P \geq 60\%$: Redemption is 150.70% 2) If $P < 60\%$: Redemption is P where: - $P = \frac{INDEX_{final}}{INDEX_{initial}}$ - $INDEX_{final}$: The closing price of the Underlying on the Final Valuation Date - $INDEX_{init}$: The closing price of the Index on the Fixing Date

Table D.0.1: Rabo Deep Autocall Certificate